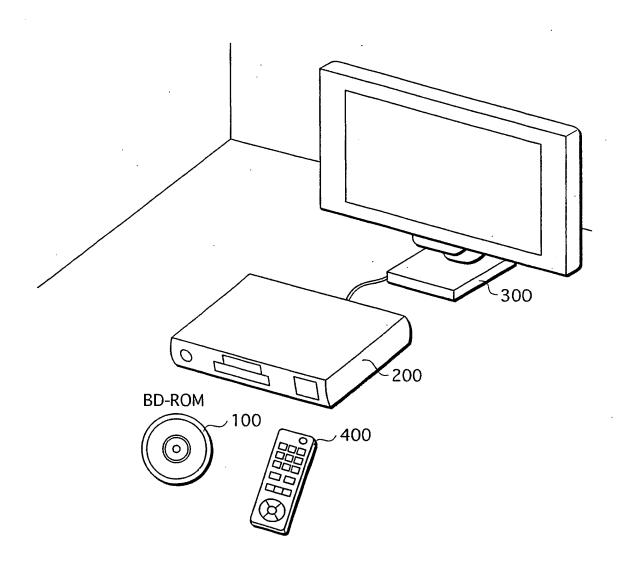
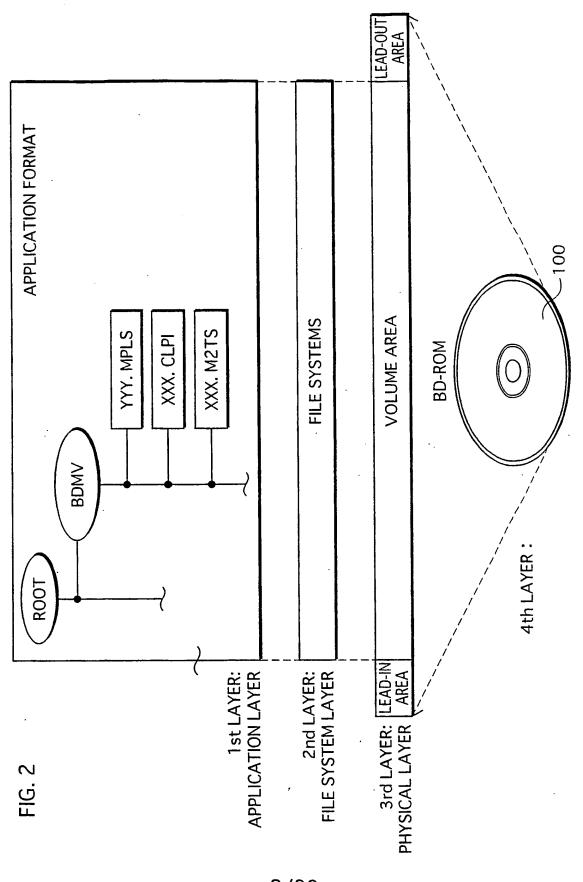
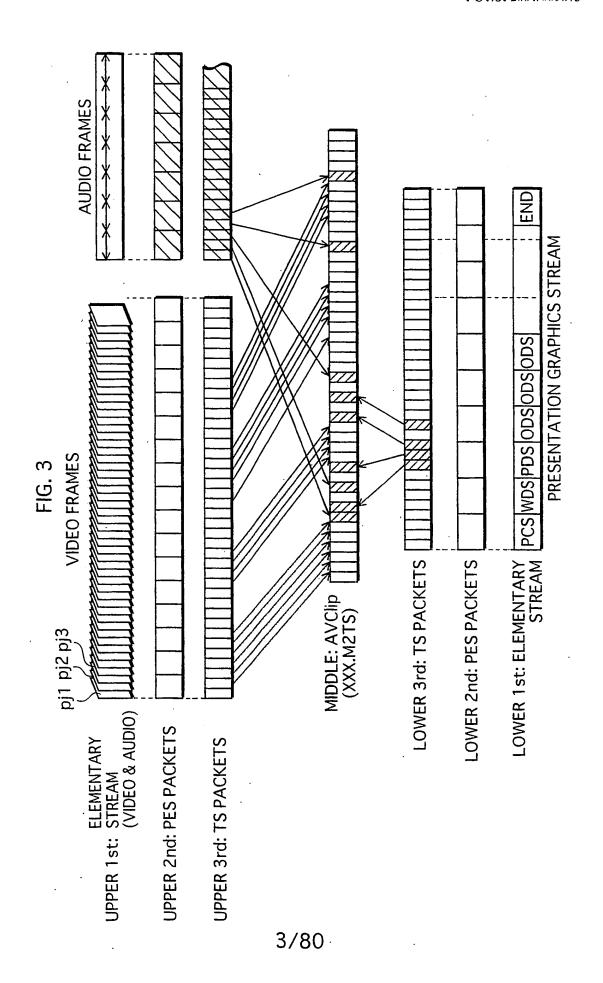
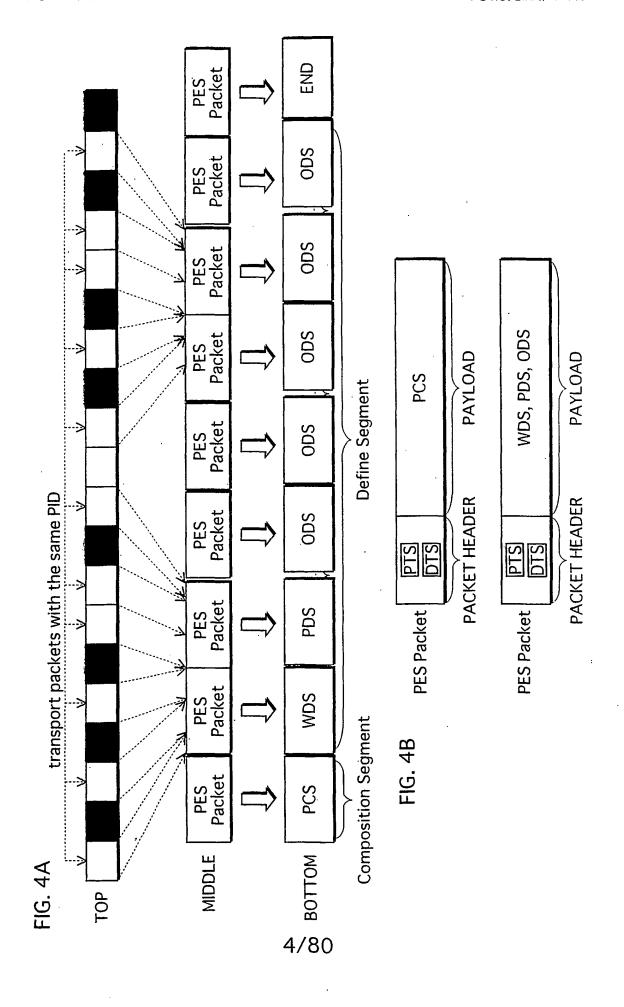
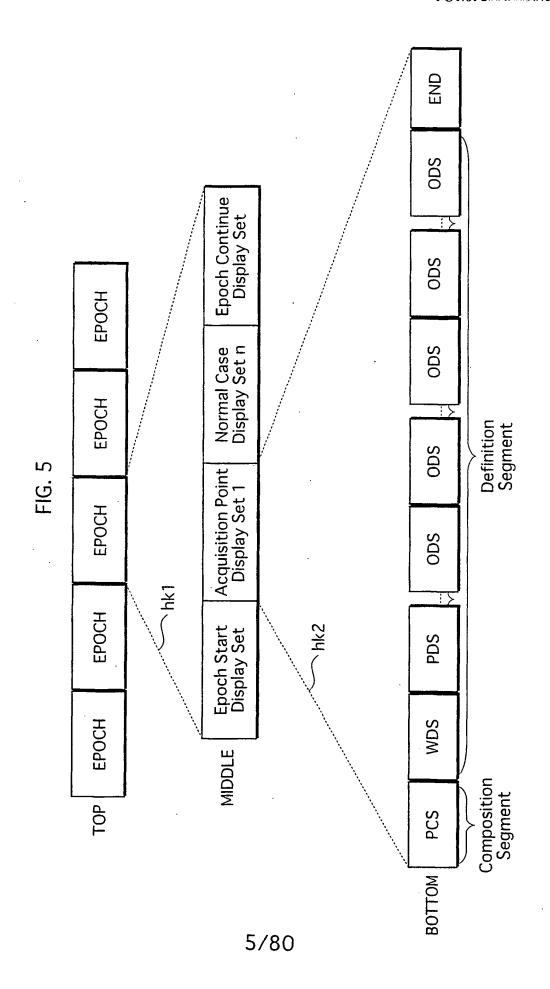
FIG. 1











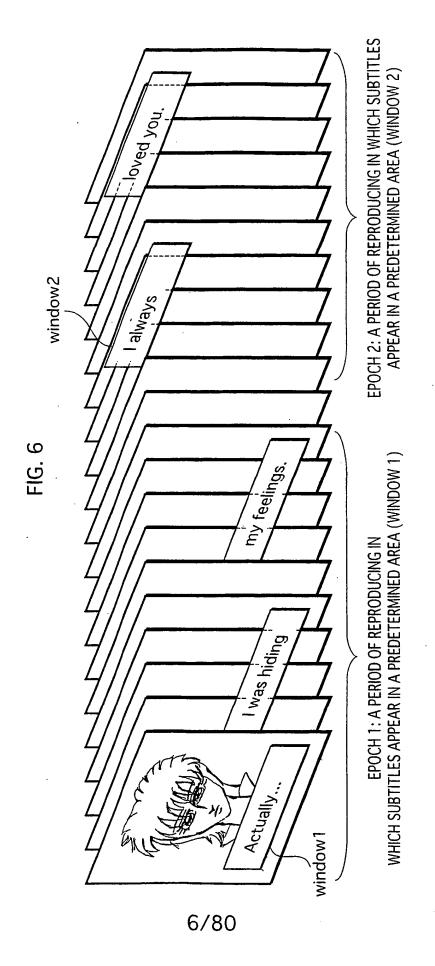


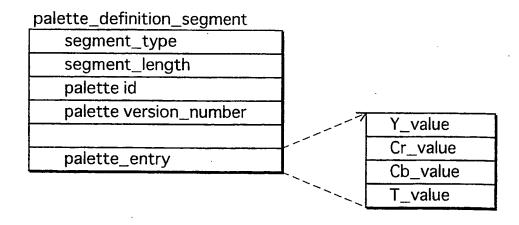
FIG. 7A

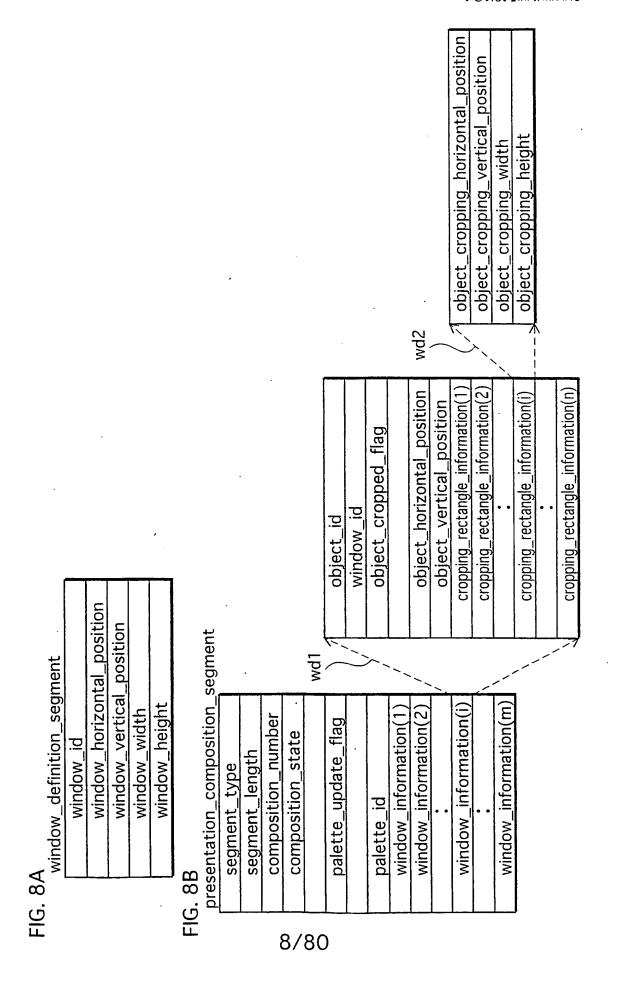
object_definition_segment
segment_type
segment_length
object_id
object_version_number
last in sequence flag

object_data_fragment

COMPRESSED GRAPHICS OBJECT

FIG. 7B





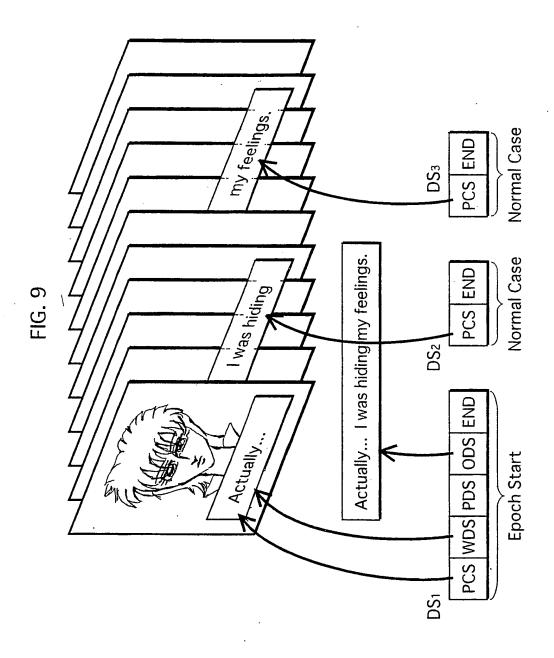


FIG. 10

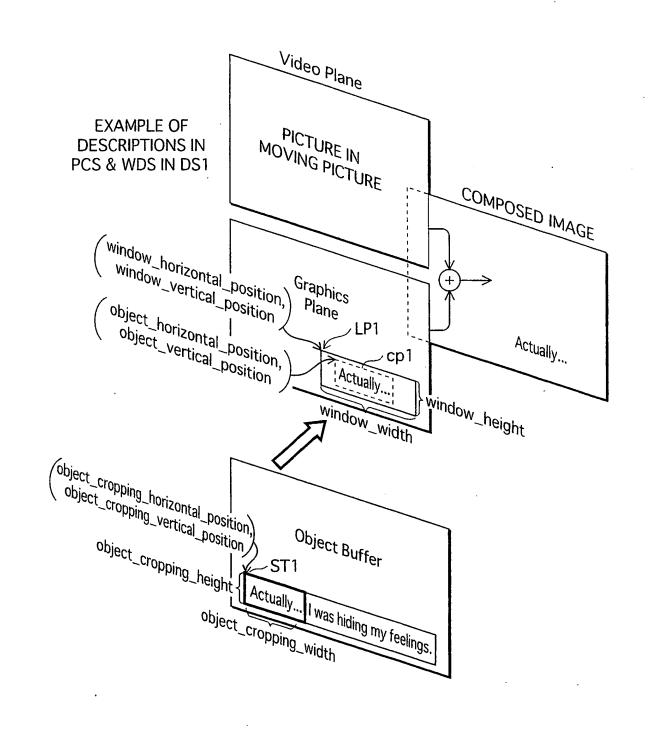


FIG. 11

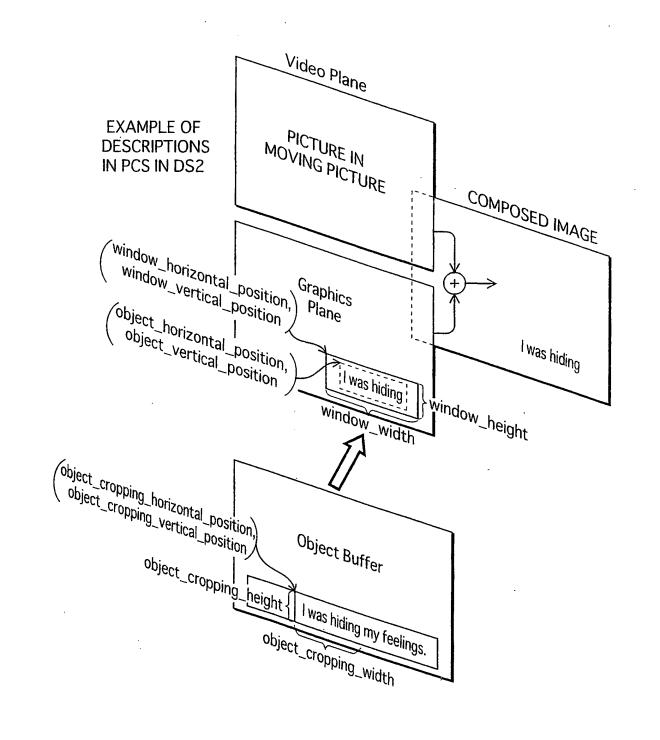
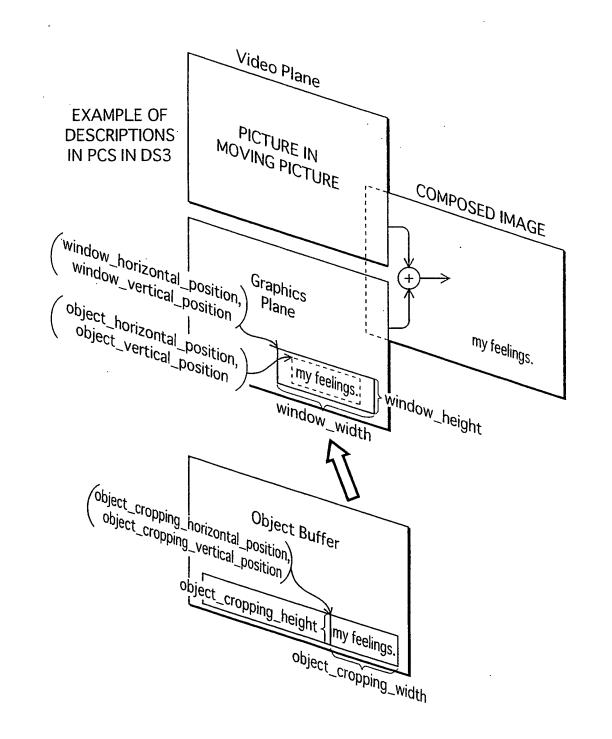
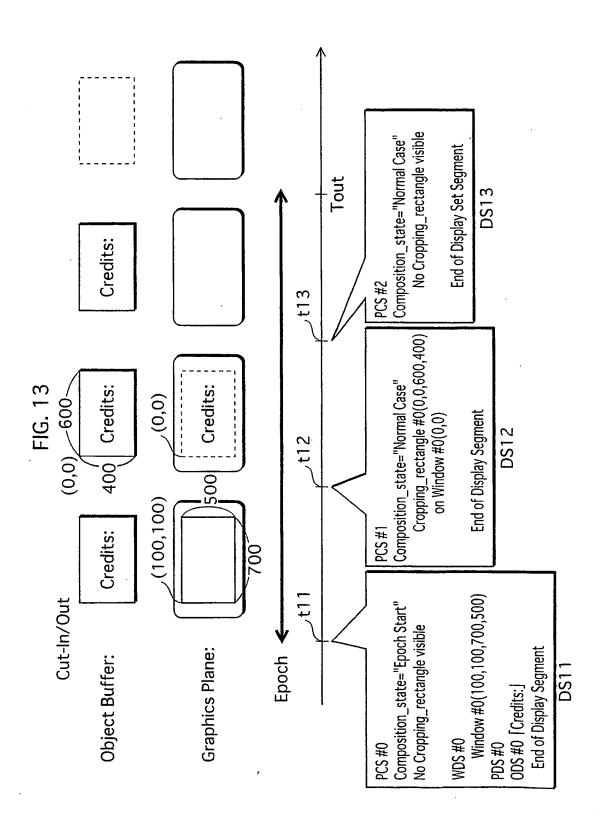
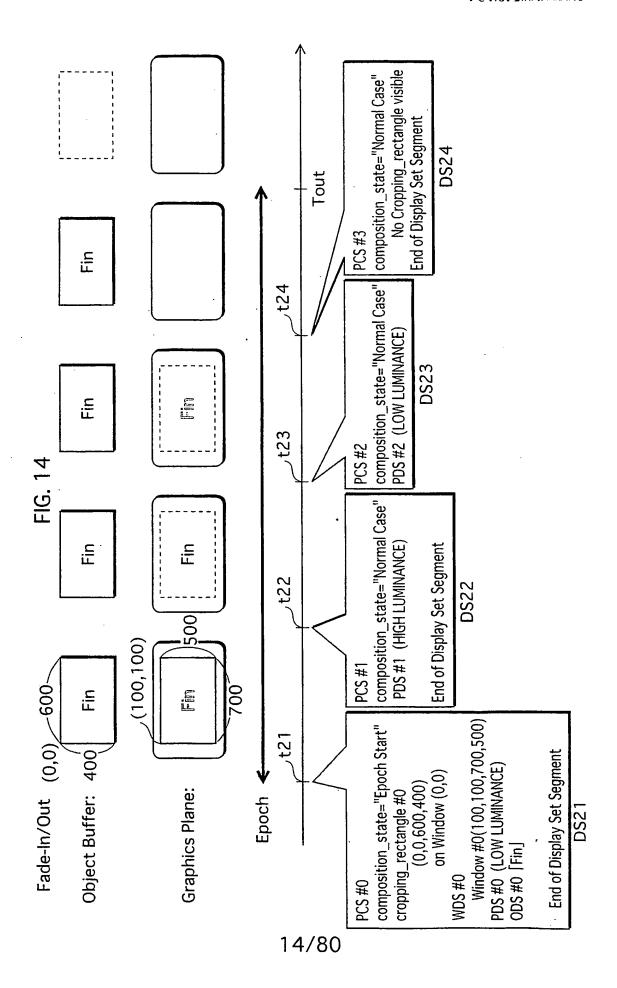
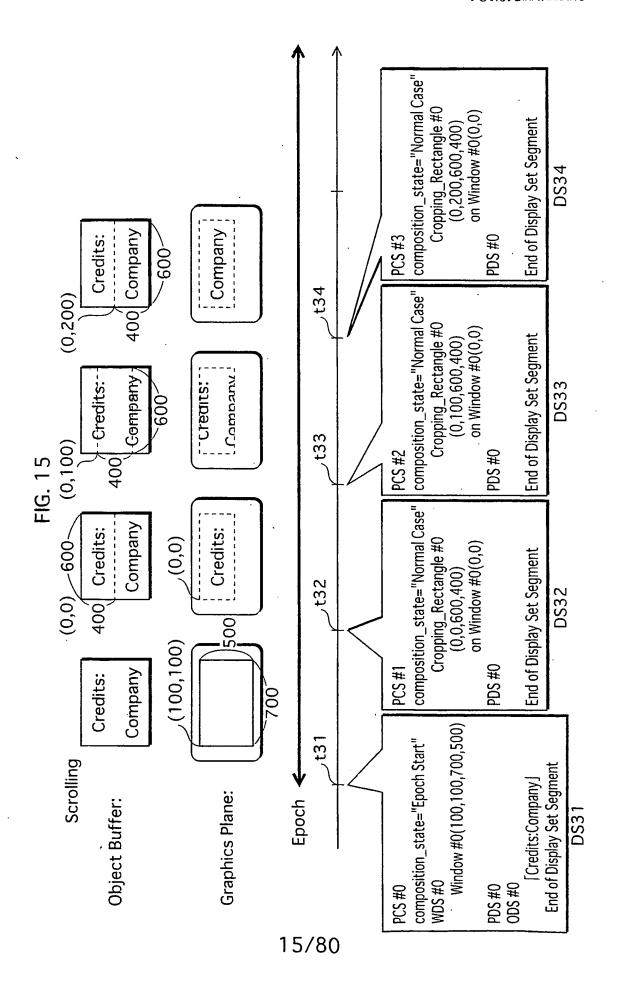


FIG. 12









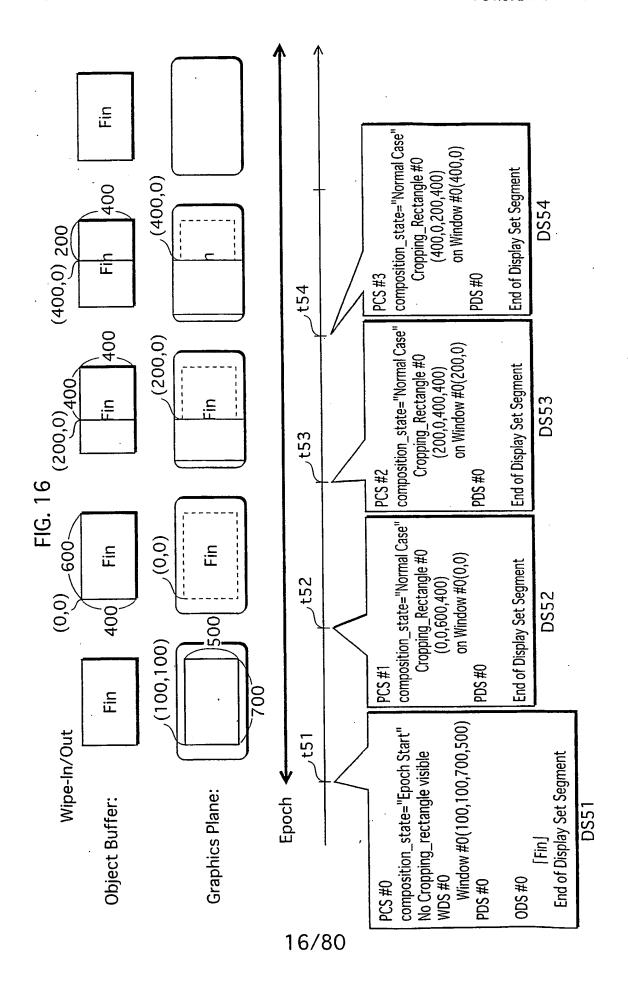
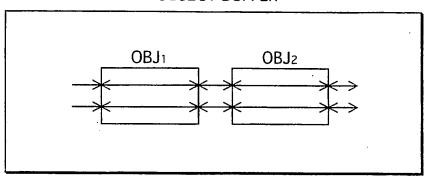
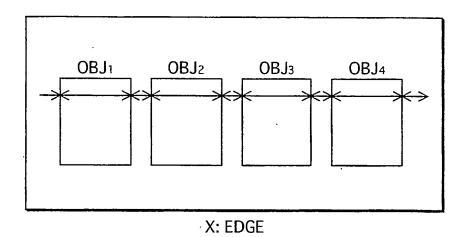


FIG. 17

OBJECT BUFFER





```
FIG. 18
          PTS( DSn[PCS)] >=DTS( DSn[PCS] )+DECODEDURATION( DSn )
 Where:
        DECODEDURATION(DSn) is calculated as follows:
    decode duration = 0:
    decode duration += PLANEINITIALIZATIONTIME( DSn );
    if( DSn. PCS. num_of_objects == 2 )
        decode_duration += WAIT( DSn, DSn. PCS. OBJ[0], decode_duration );
        if(DSn. PCS. OBJ[0]. window_id == DSn. PCS. OBJ[1]. window_id)
                 decode_duration += WAIT( DSn, DSn. PCS. OBJ[1], decode_duration );
                 decode_duration += 90000*(SIZE(DSn. PCS. OBJ[0]. window id)//256*10^6):
        else
                 decode_duration += 90000*(SIZE(DSn. PCS. OBJ[0]. window_id)//256*10^6);
                 decode_duration += WAIT( DSn, DSn. PCS. OBJ[1], decode_duration );
                 decode_duration += 90000*(SIZE(DSn. PCS. OBJ[1]. window_id)//256*10^6);
    else if (DSn. PCS. num of objects == 1)
        decode duration += WAIT( DSn, DSn. PCS. OBJ[0], decode duration);
        decode_duration += 90000*( SIZE( DSn. PCS. OBJ[0]. window id )//256*10<sup>6</sup> );
    return decode duration:
        PLANEINITIALIZATIONTIME( DSn ) is calculated as follows:
    initialize duration=0:
    if( DSn. PCS. composition state= = EPOCH START )
       initialize_duration = 90000*( 8*video_width*video_height//256*10<sup>6</sup>);
    else
        for (i=0; i < WDS. num windows; i++)
                if( EMPTY(DSn.WDS.WIN[i],DSn ) )
                      initialize duration += 90000*(SIZE(DSn. WDS. WIN[i])//256*10^6);
    return initialize_duration;
        WAIT( DSn, OBJ, current_duration ) is calculated as follows:
    wait_duration = 0;
    if(EXISTS(OBJ. object id, DSn))
        object_definition_ready_time = PTS( GET( OBJ. object_id. DSn ) );
        current_time = DTS( DSn. PCS )+current duration;
        if (current time < object definition ready time)
                wait_duration += object_ definition ready time - current time );
    return wait_duration;
```

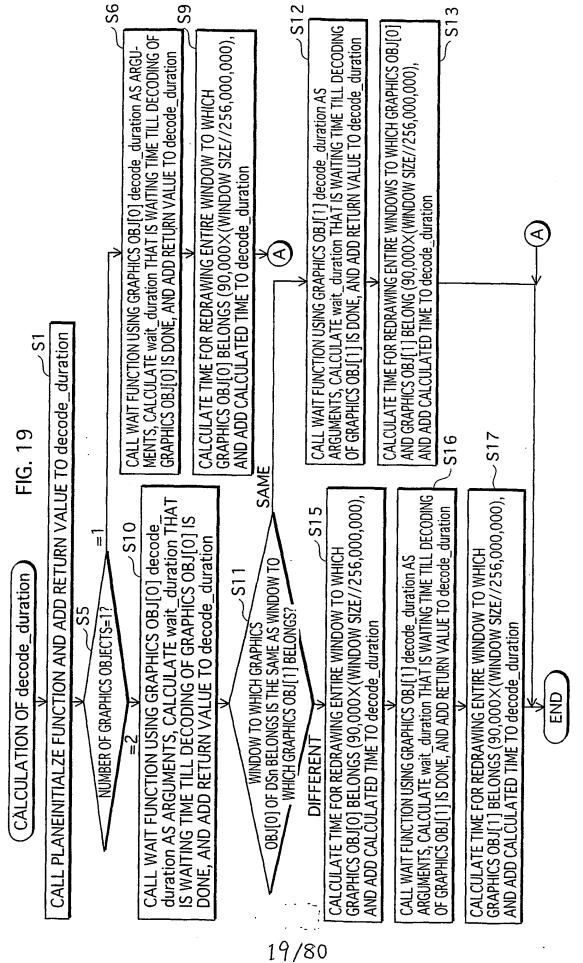


FIG. 20A PLANE INITIALIZATION TIME *,*S2 composition_state NO =Epoch_Start? YES /S3 **S4** initialize _←TIME NECESSARY TO CLEAR initialize_←TIME NECESSARY TO CLEAR duration **ALL WINDOWS GRAPHICS PLANE** duration Σ (90,000×(SIZE OF (90,000×(GRAPHICS WINDOW[i] //256,000,000) PLANE SIZE//256,000,000) *i MAY BE 0 OR 1 RETURN initialize duration

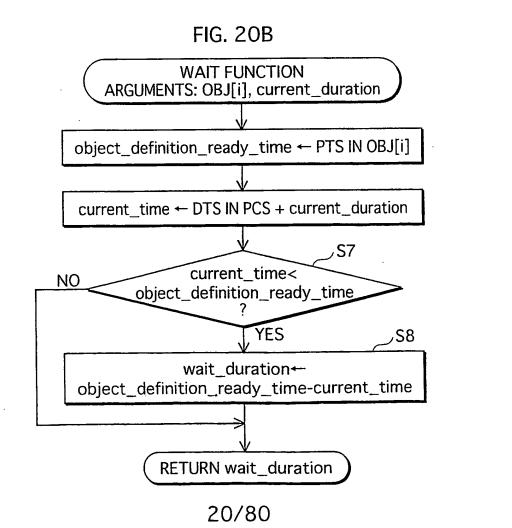
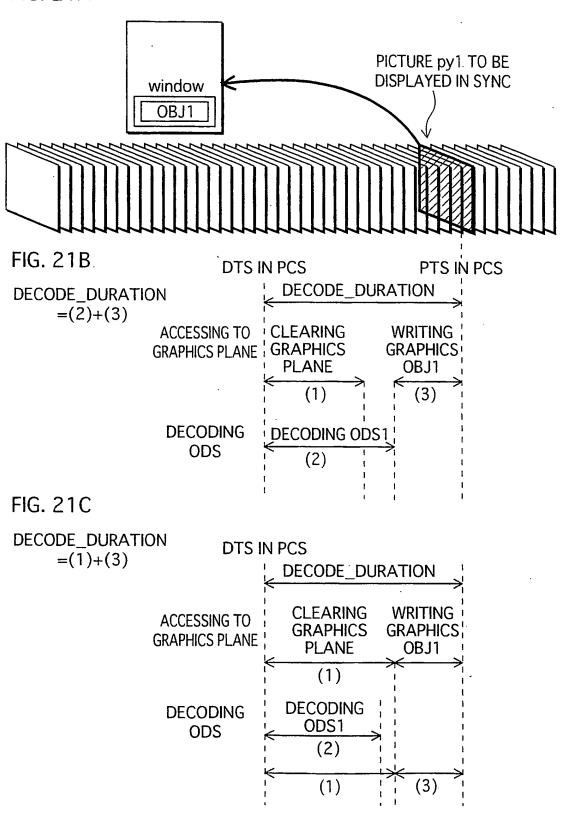
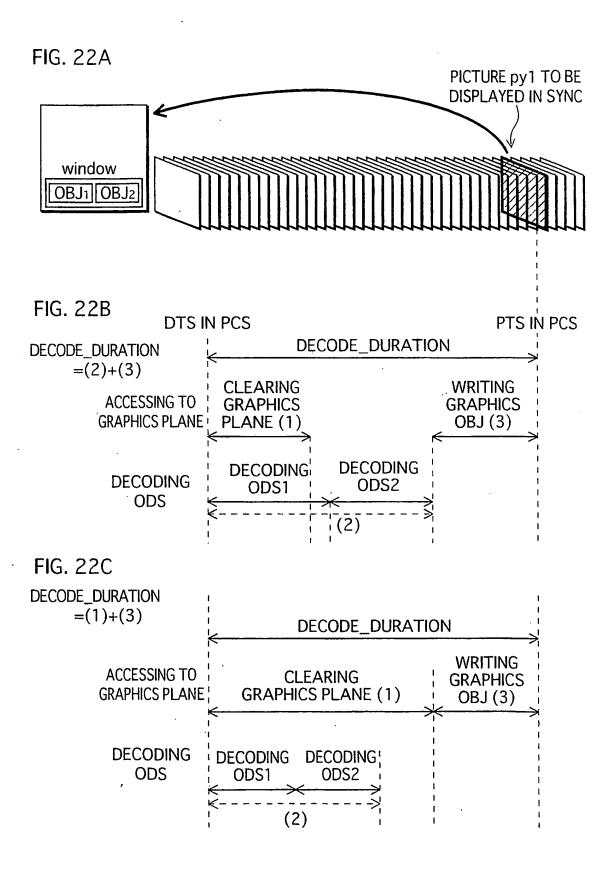
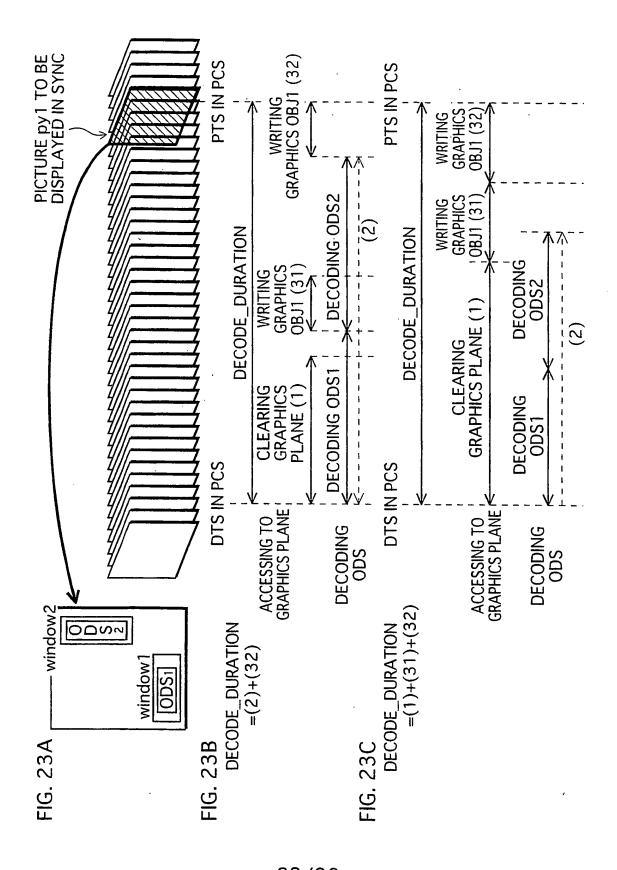
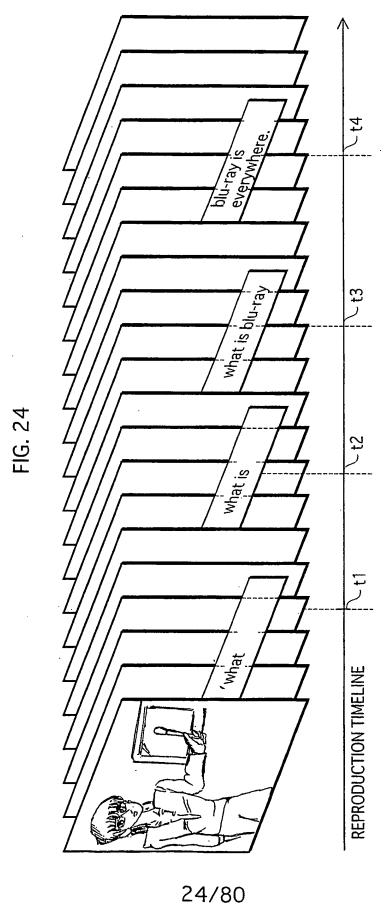


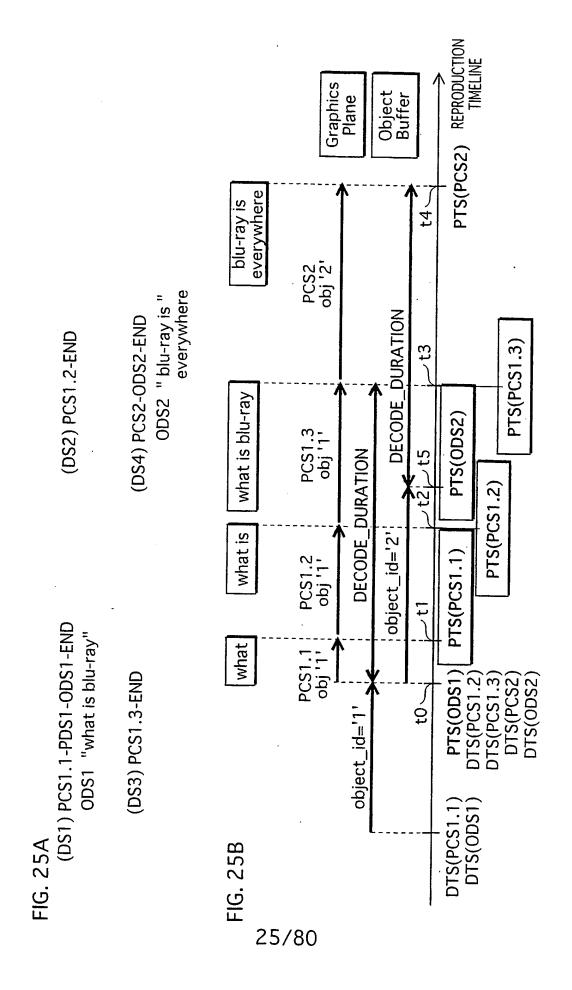
FIG. 21A

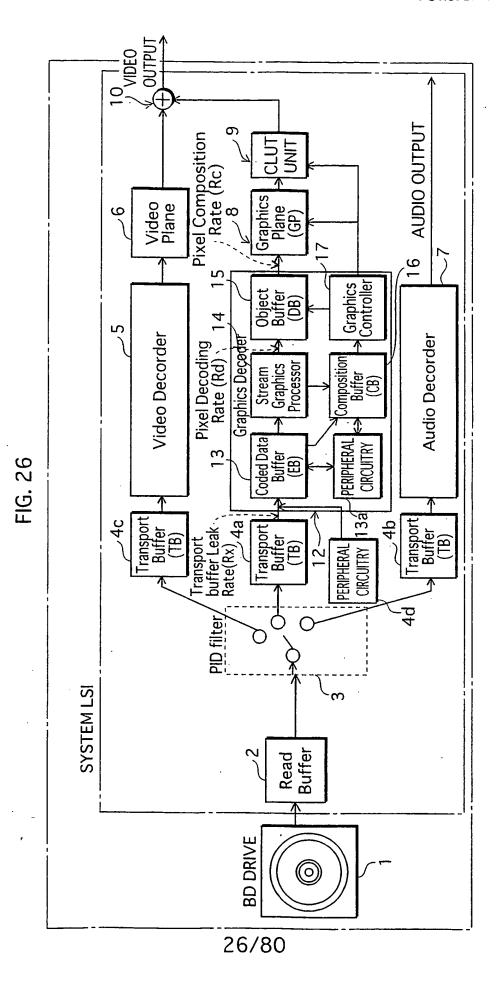


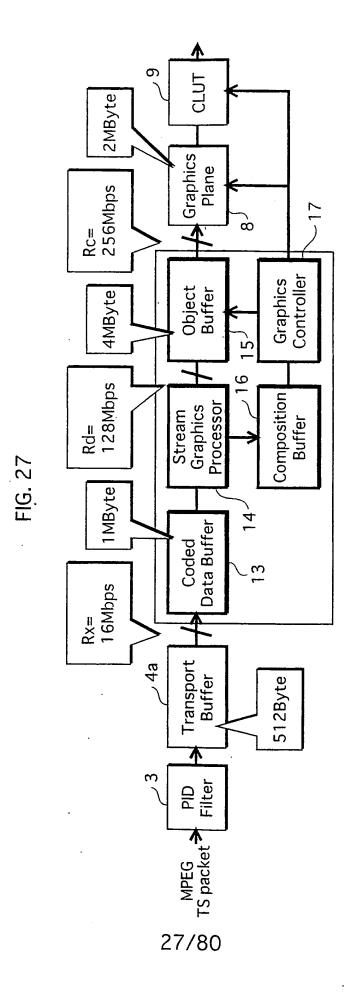


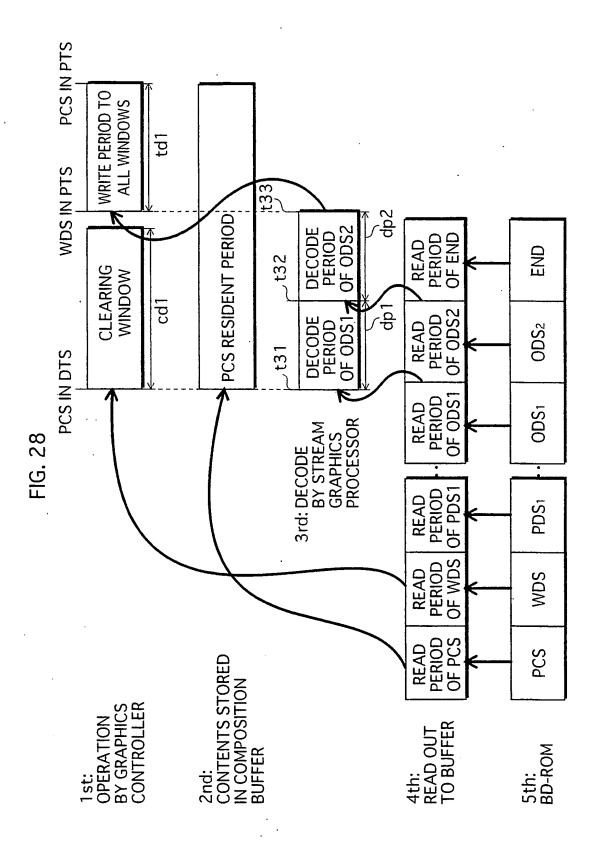


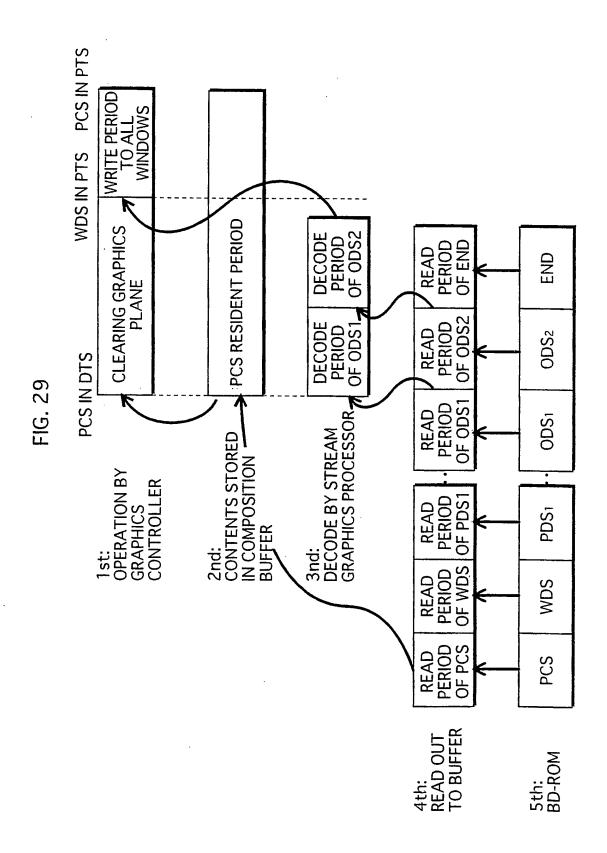


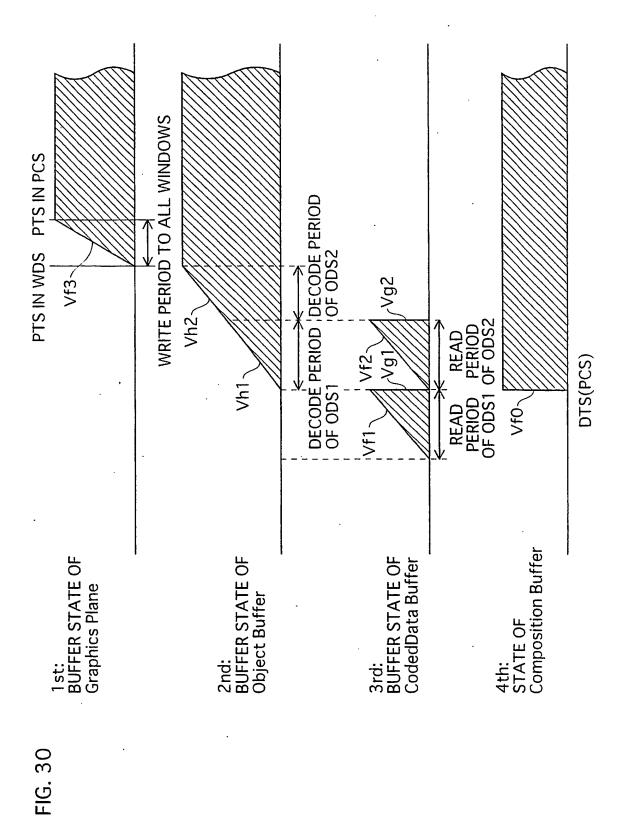




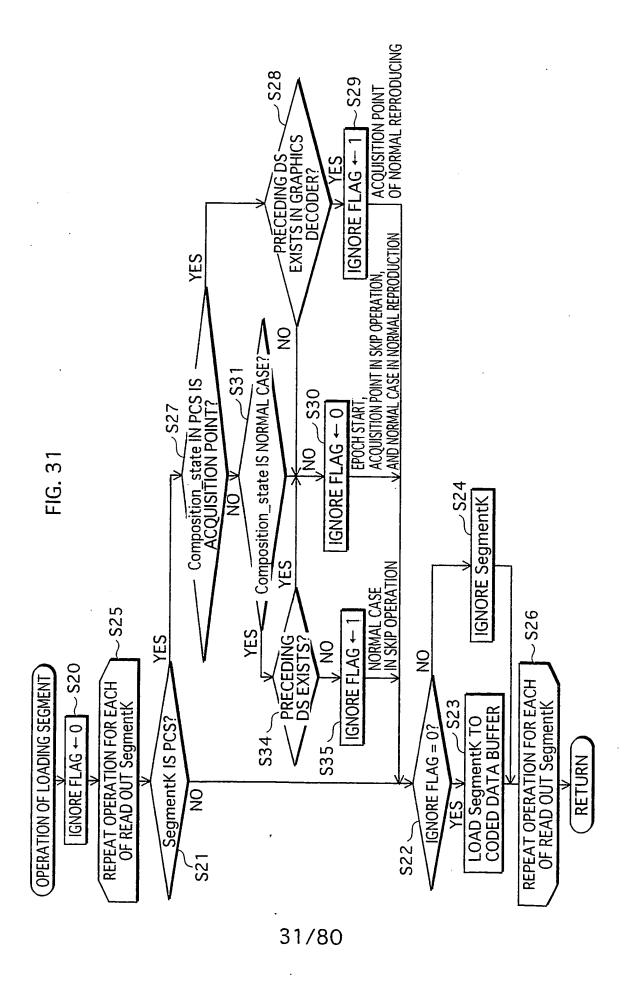


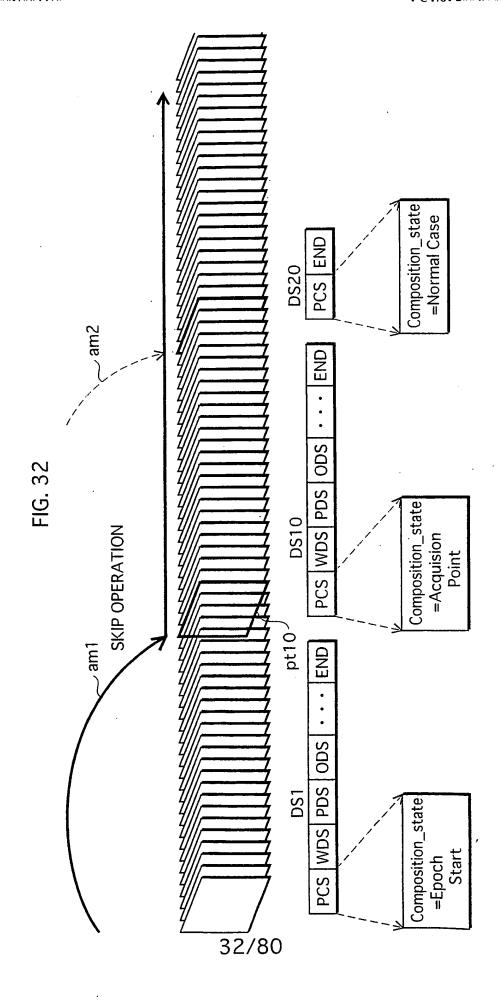


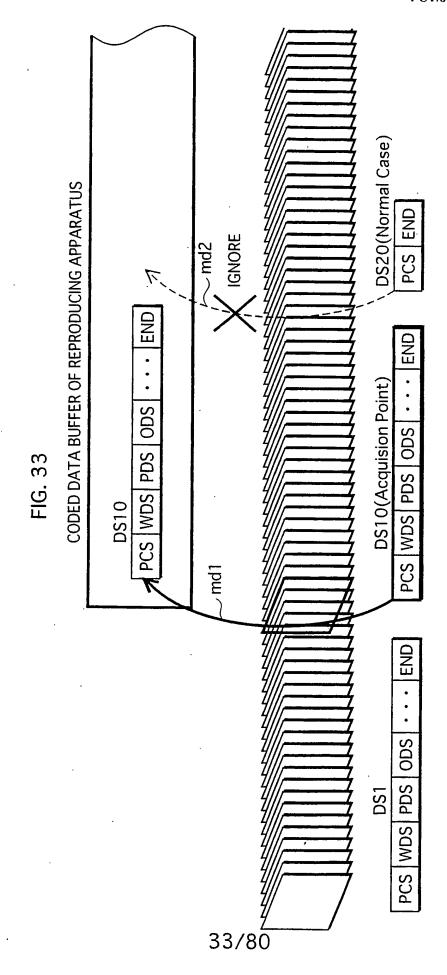


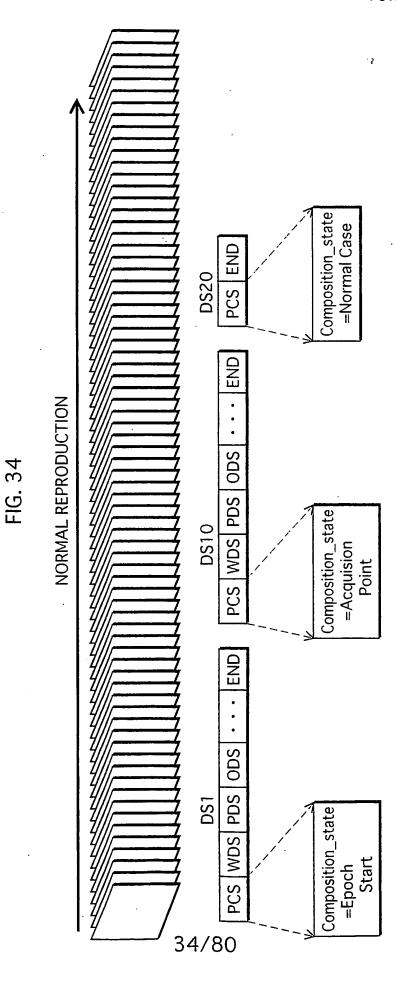


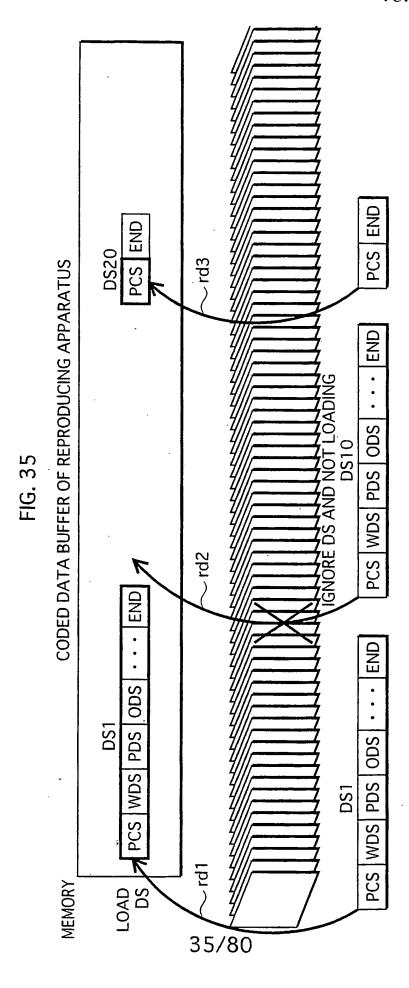
30/80

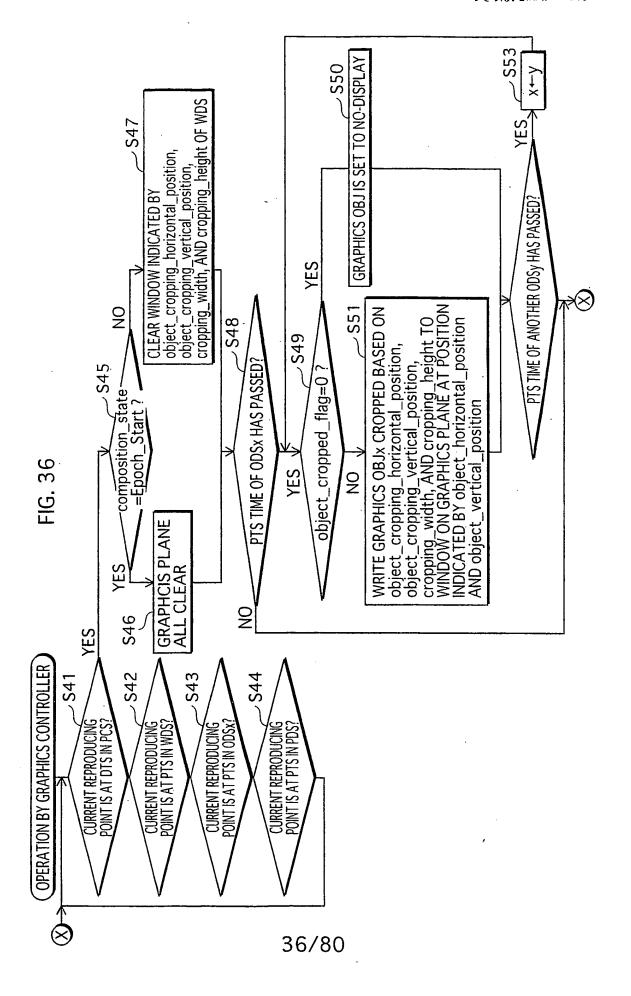


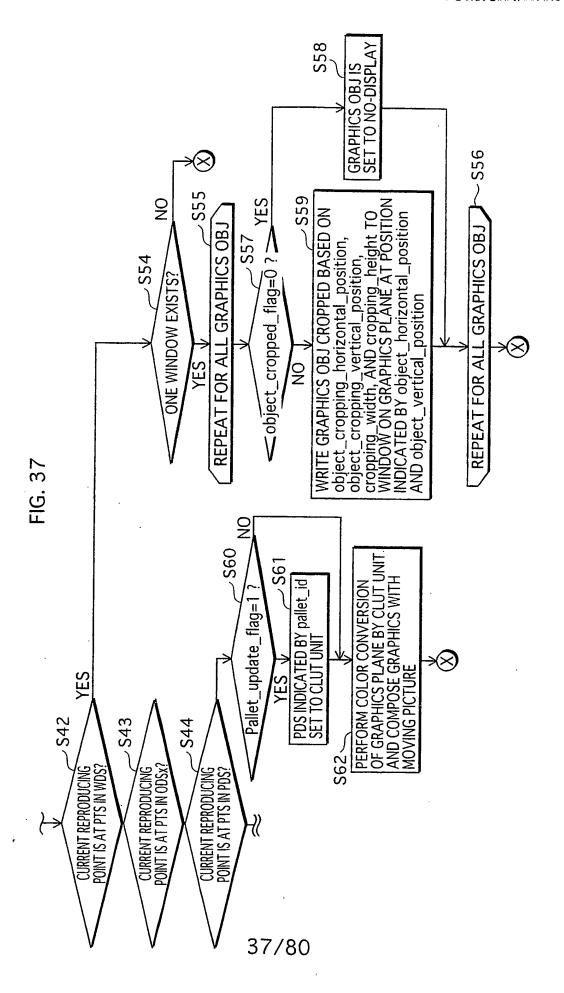












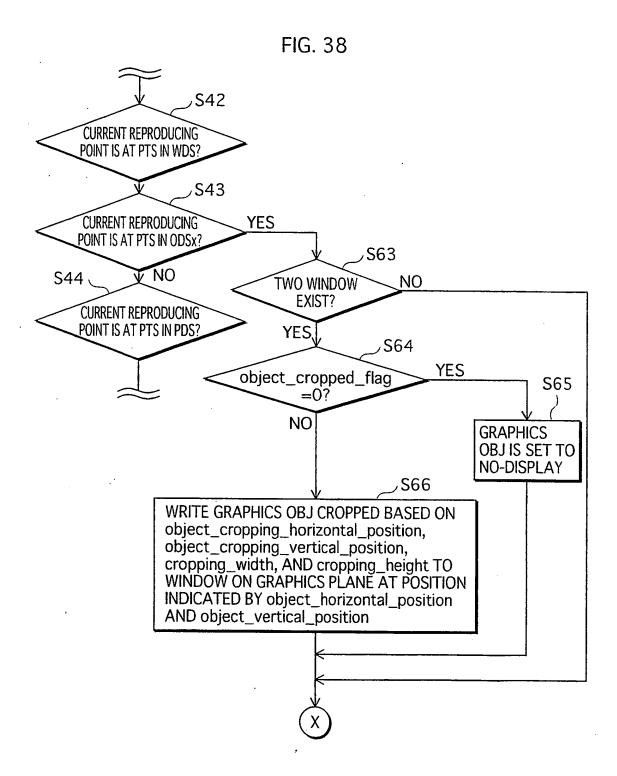
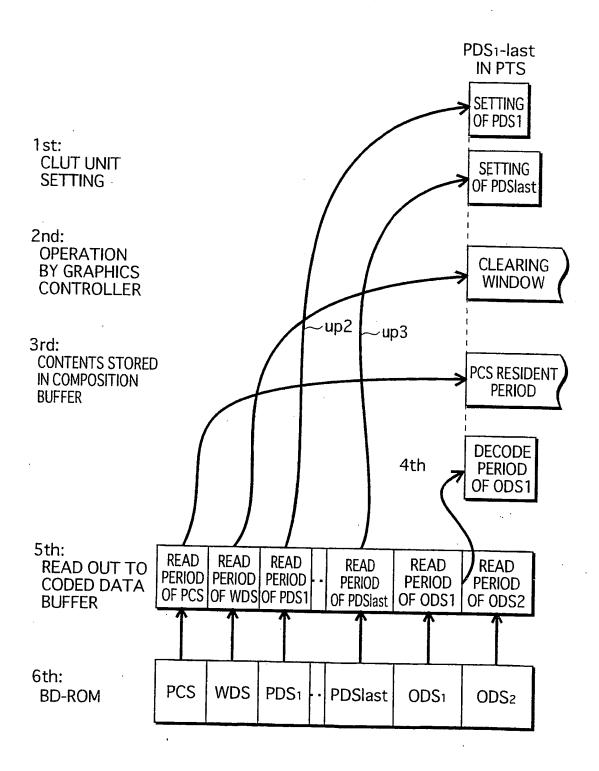
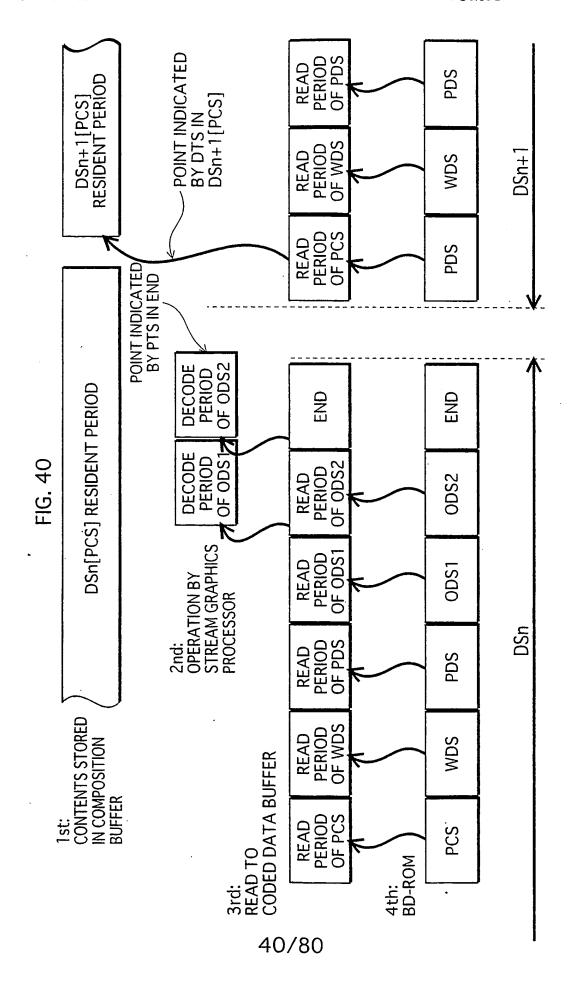


FIG. 39





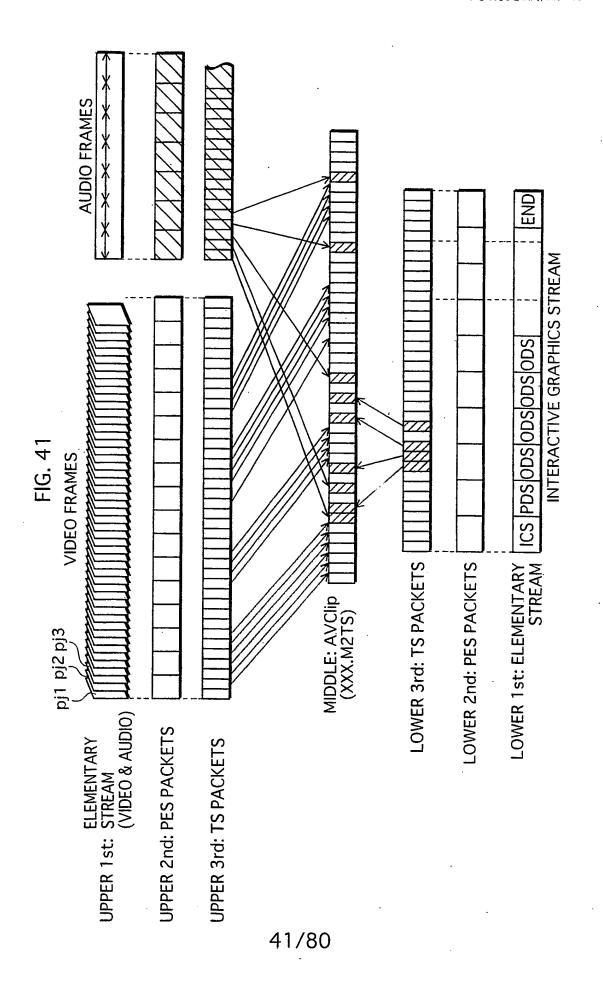


FIG.42A

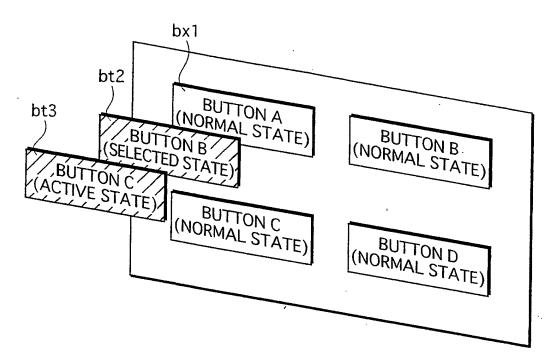
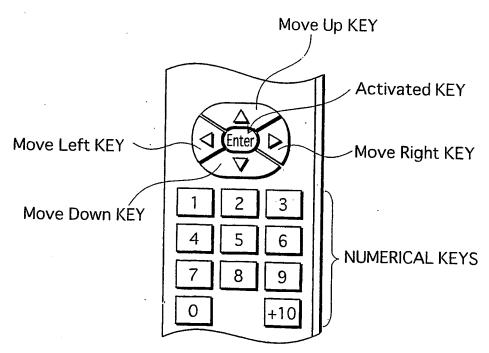
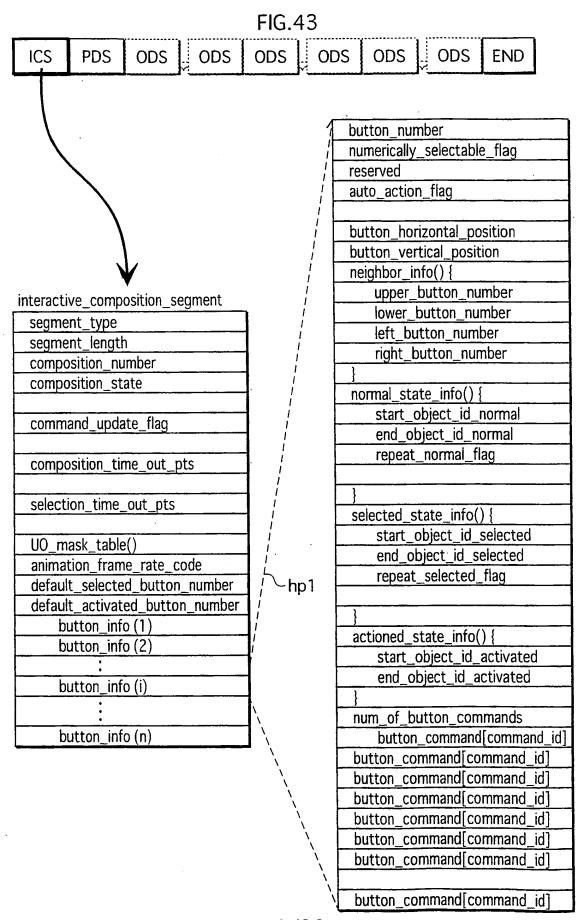
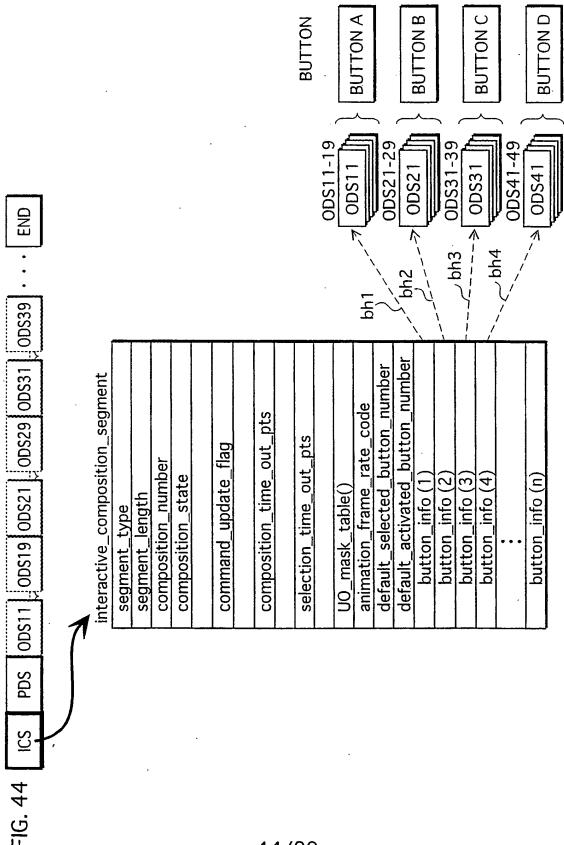


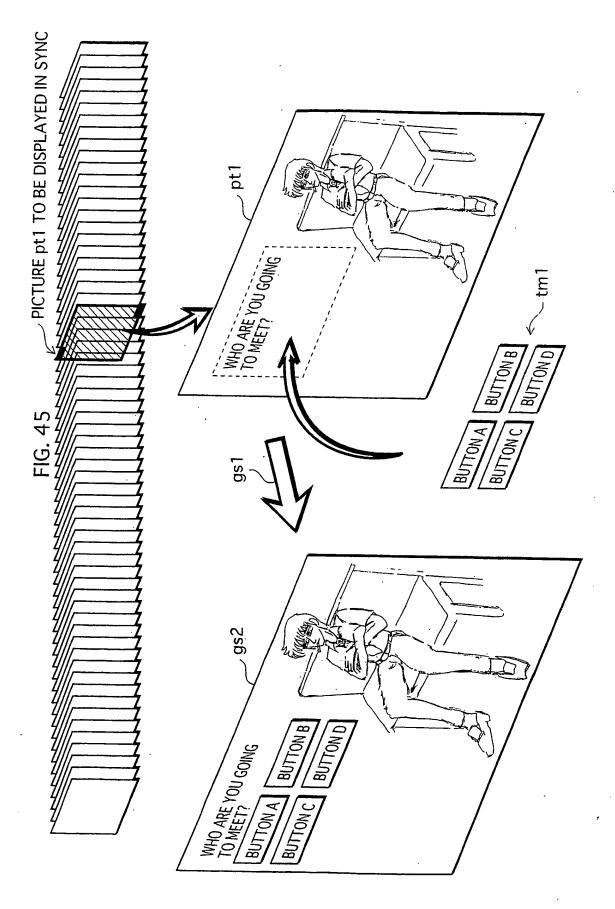
FIG.42B





43/80

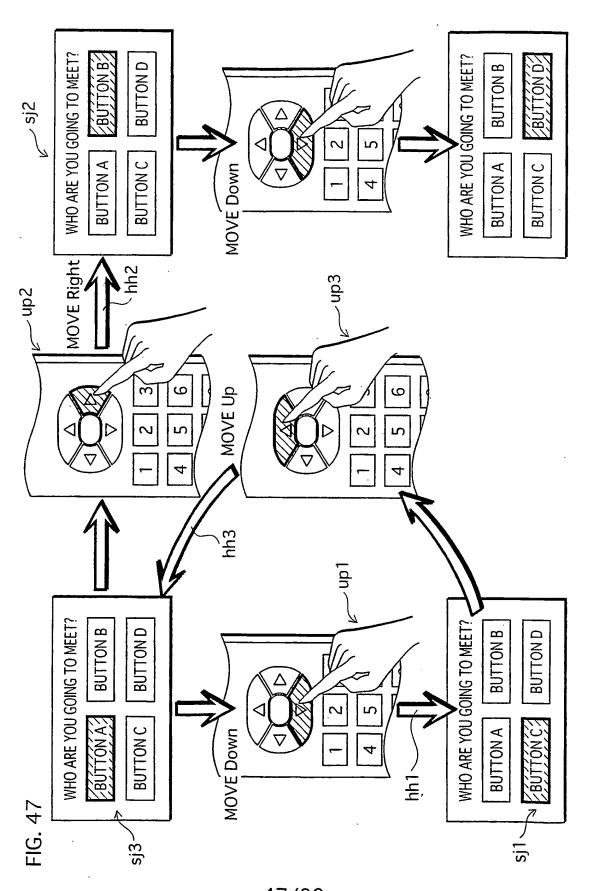




45/80

FIG.46

	neighbor_info()	
	upper_button_number	· · · BUTTON A
į.	lower_button_number	· · · BUTTON C
Í	left_button_number	· · · BUTTON A
;	right_button_number	··· BUTTON B
1	normal_state_info()	
i	start_object_id_normal	· · · ODS11
į	end_object_id_normal	··· ODS13
i	repeat_normal_flag	
1	selected_state_info()	
İ	start_object_id_selected	· · · ODS14
interactive	end_object_id_selected	ODS16
composition segment	repeat_selected_flag	
	actioned_state_info()	
segment_type	start_object_id_activated	· · · ODS17
segment_length	end_object_id_activated	· · · ODS19
composition_number		(330,0)
composition_state	neighbor_info()	1
hutton info (1)	upper_button_number	··· BUTTON A
button_info(1) //	lower button number	BUTTON C
button_info (2)	left_button_number	··· BUTTON C
button_info (3)	right_button_number	BUTTON D
button_info (4)	normal_state_info()	BUTTOND
		ODC21
button_info (n)	start_object_id_normal end_object_id_normal	ODS31
STATE CONTROL		··· ODS33
INFORMATION	repeat_normal_flag	
	selected_state_info()	00004
	start_object_id_selected	ODS34
1	end_object_id_selected	··· ODS36
Ì	repeat_selected_flag	
	actioned_state_info()	[0000=]
1	start_object_id_activated	· · · ODS37
\	end_object_id_activated	··· ODS39



47/80

FIG.48

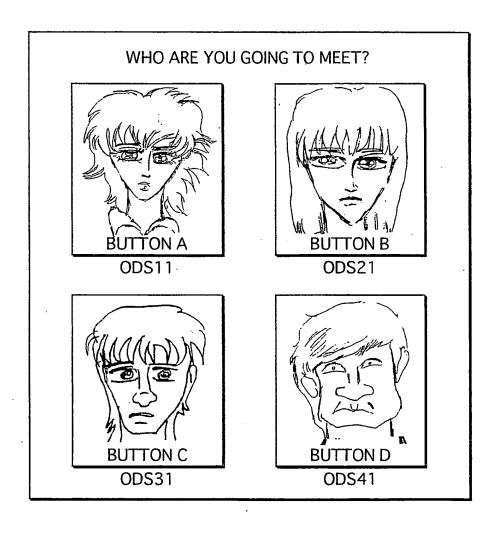
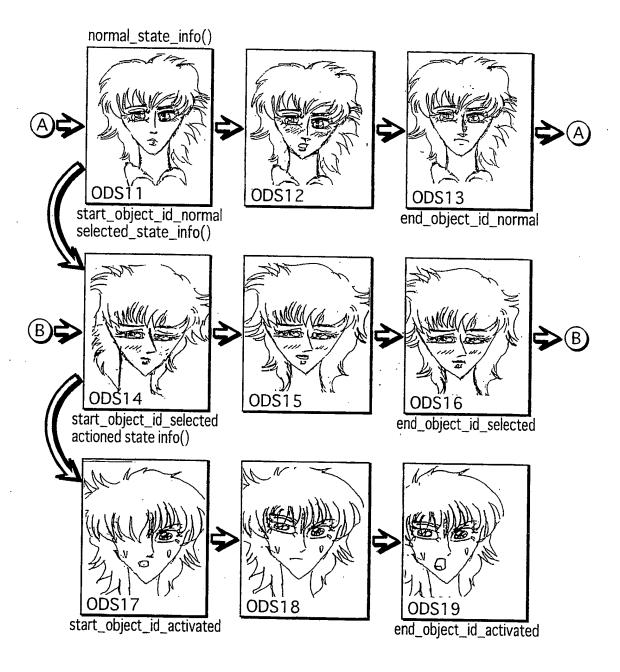
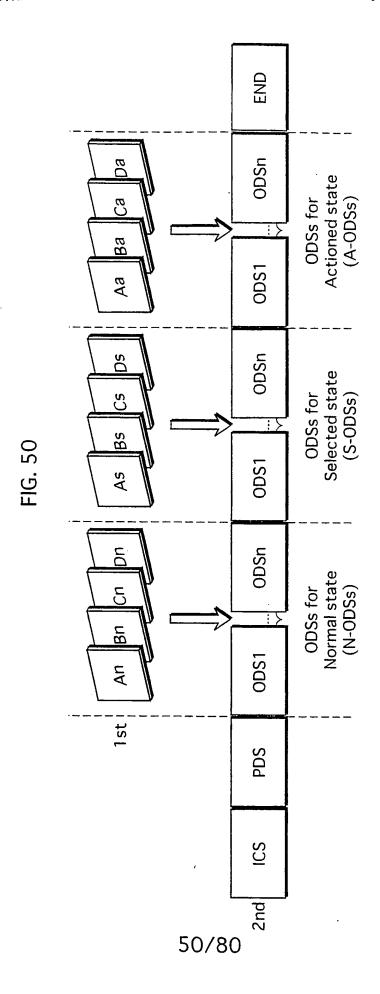
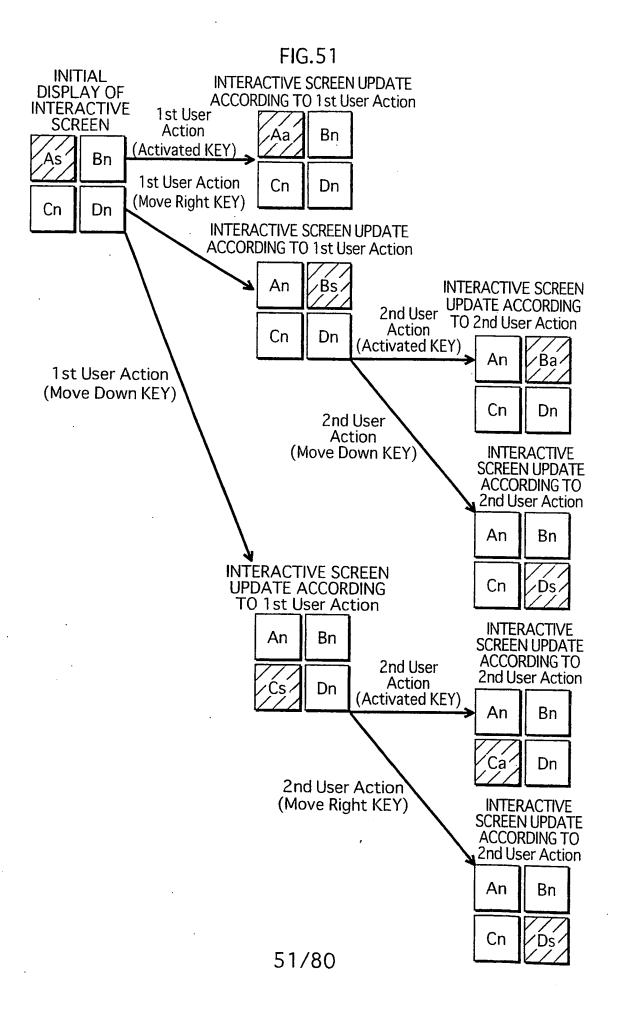
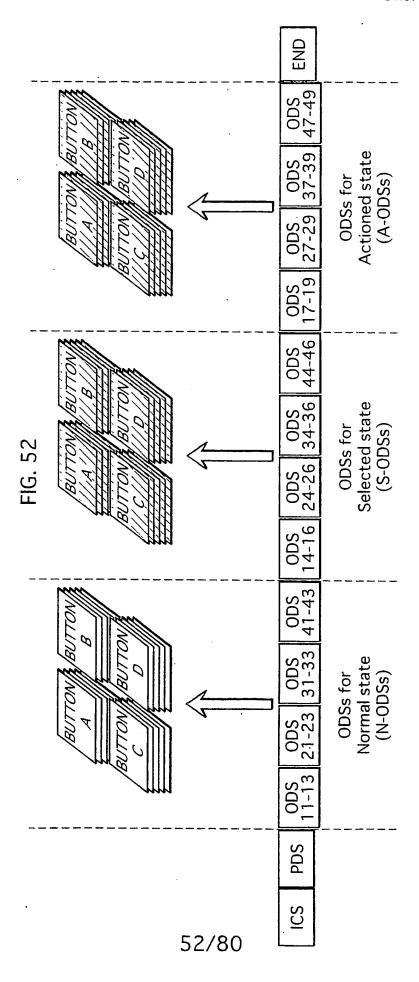


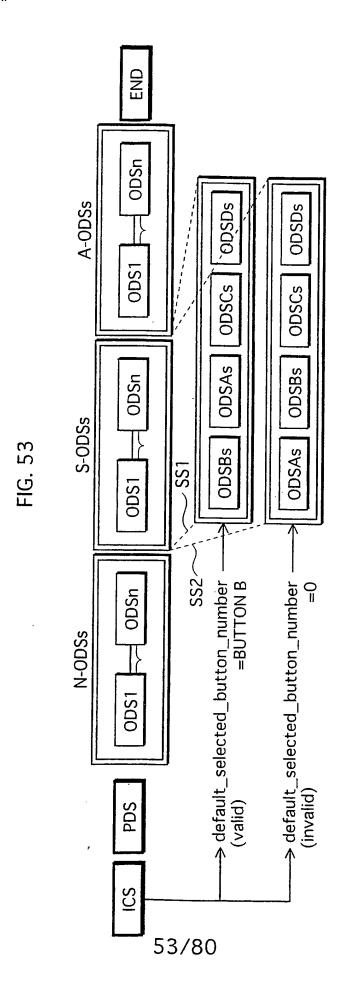
FIG.49









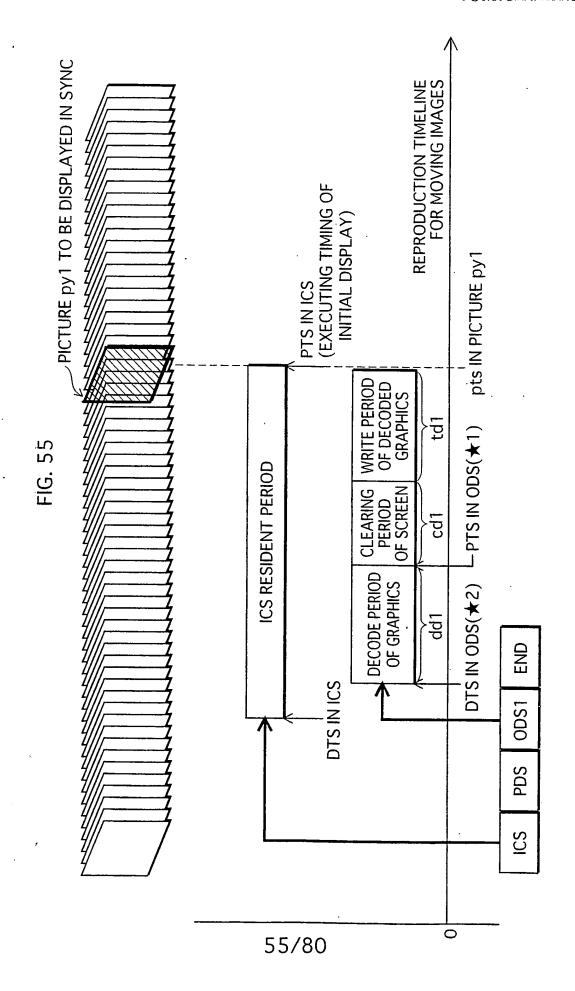


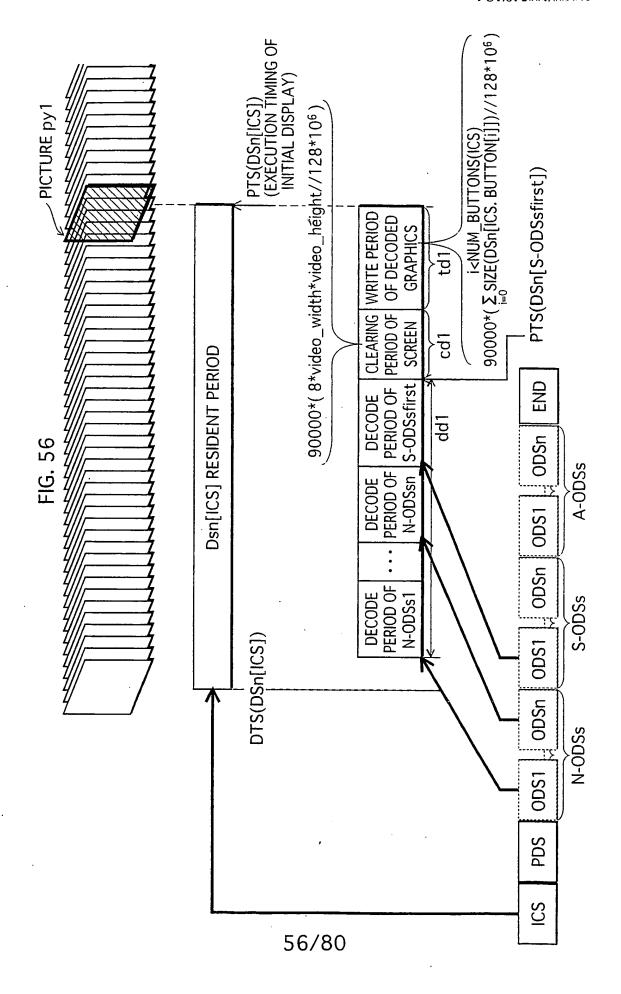
Ds2 Ds1 Cs2 Bs2 ... Cs1 S-ODSs As2 |··· Bs1 As1 Dn2 <u>P</u> default_selected_button_number is indicated Cn2 ... Bn2 ... Cn1 N-ODSs An2 ... Bn1 FIG. 54A An1

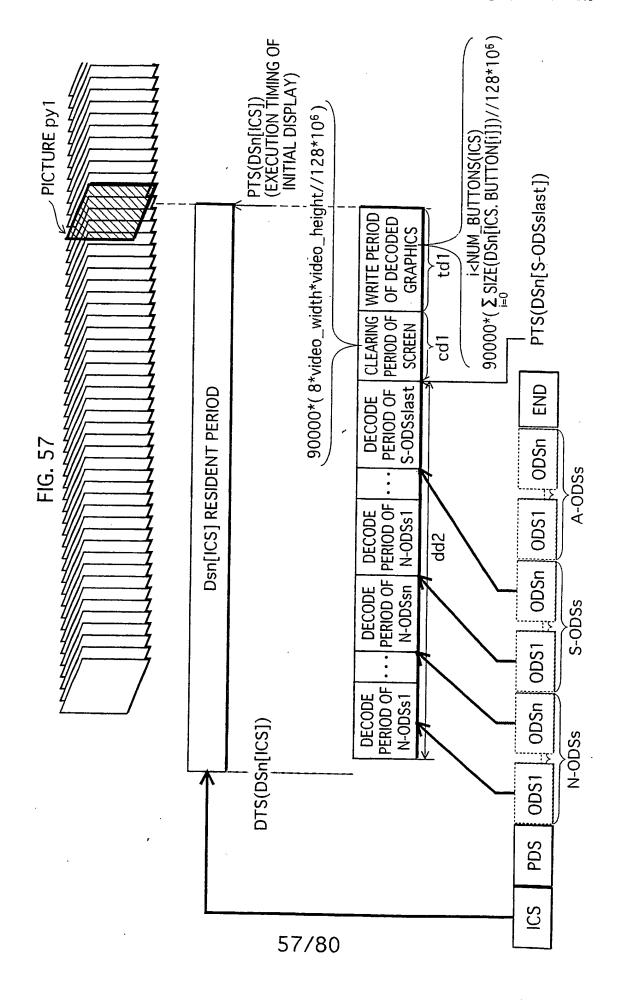
\$\infty \text{SIZE(DSn[ICS.BUTTON[i]]} = \size(As1) + \size(\text{Bn1}) + \size(\text{Cn1}) + \size(\text{Cn1}) + \text{Size(Dn1)}

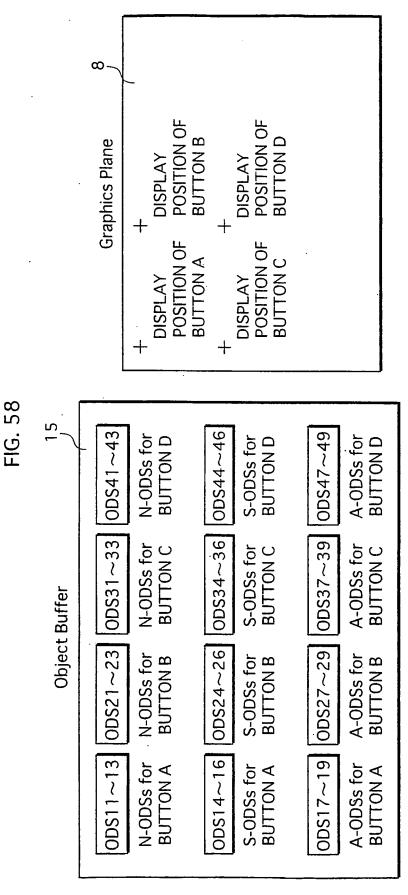
+max(size(Dn1),size(Ds1)) Ds2 +max(size(Bn1),size(Bs1))
+max(size(Cn1),size(Cs1) Σ SIZE(DSn[ICS.BUTTON[i]]=max(size(An1),size(As1)) ... Ds1 Cs2 Bs2 ... Cs1 S-ODSs ... Bs1 As2 As1 Dn2 Cn2 ... Dn1 default_selected_button_number=0 Bn2 ··· Cn1 N-ODSs An2 |--- Bn1 Anl

FIG. 54B

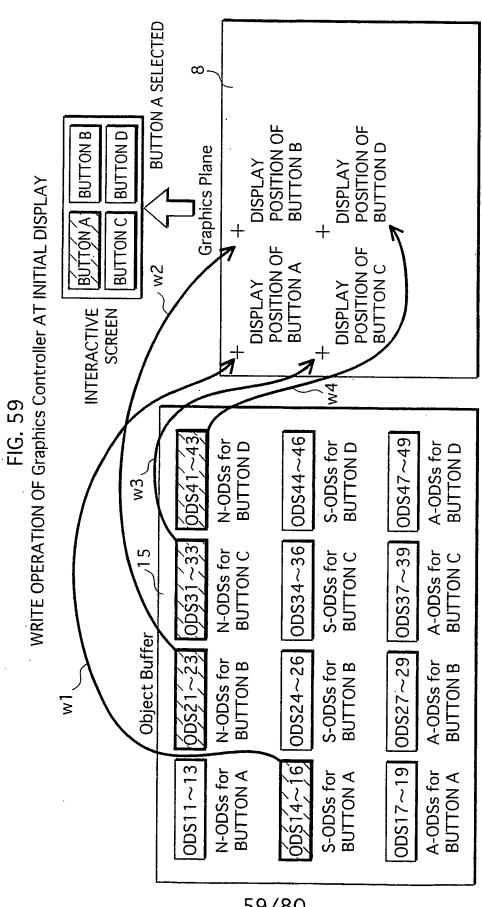






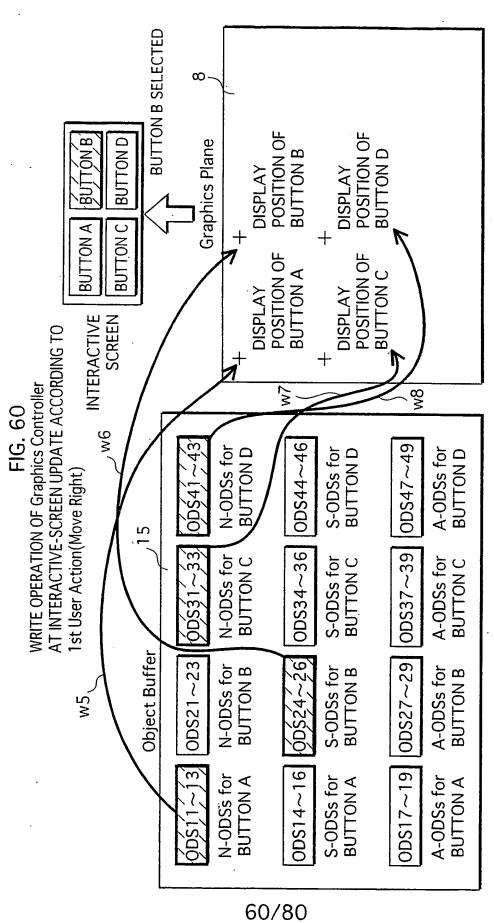


DISPLAY POSITION OF BUTTON=DISPLAY POSITION DEFINED BY button_horizontal position, button_vertical_position OF BUTTON INFORMATION

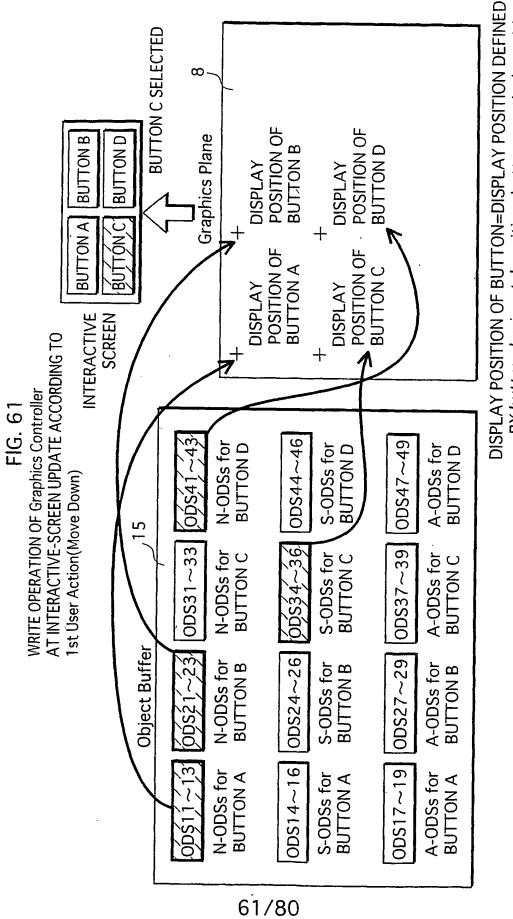


DISPLAY POSITION OF BUTTON=DISPLAY POSITION DEFINED BY button_horizontal position, button_vertical_position OF BUTTON INFORMATION

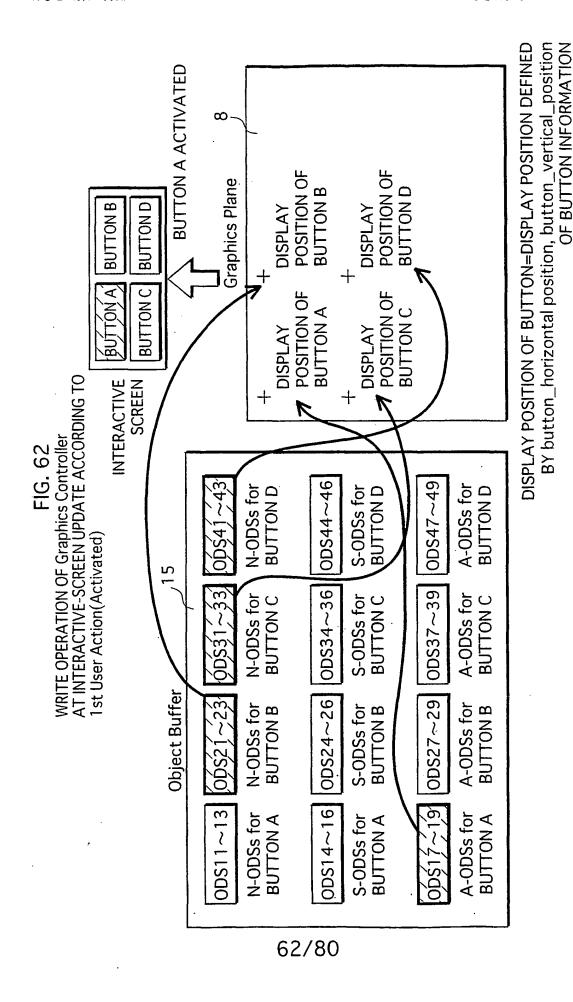
59/80

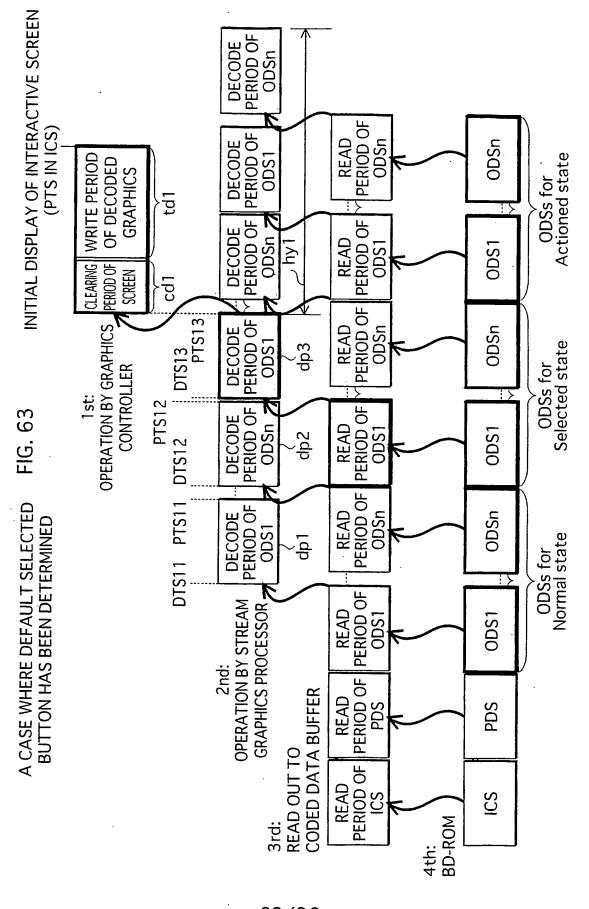


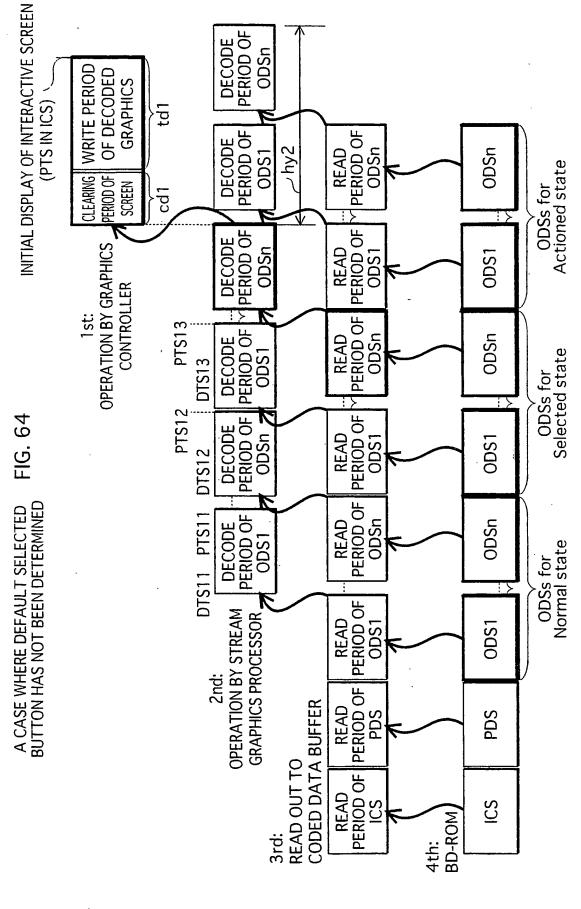
DISPLAY POSITION OF BUTTON=DISPLAY POSITION DEFINED BY button_horizontal position, button_vertical_position OF BUTTON INFORMATION



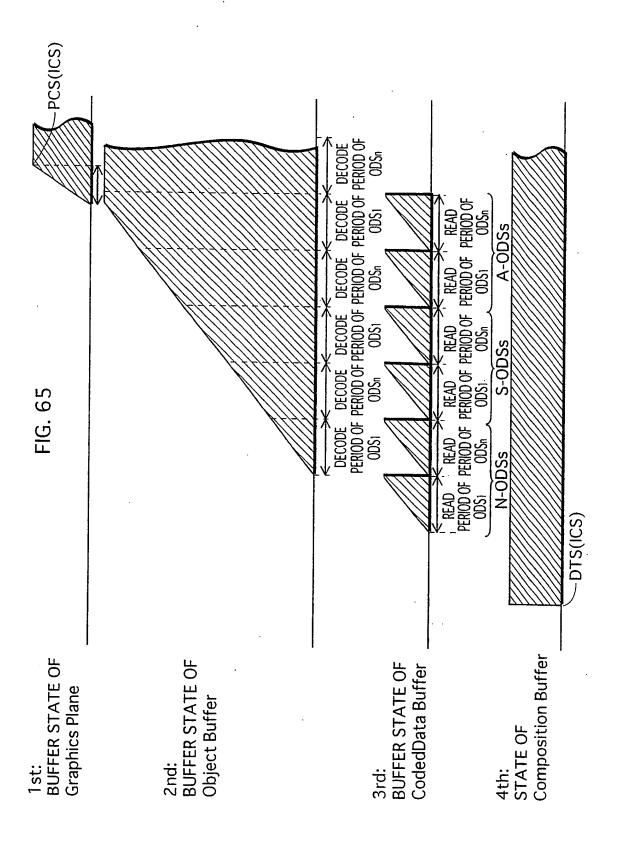
MSPLAY POSITION OF BUTTON=DISPLAY POSITION DEFINED BY button_horizontal position, button_vertical_position OF BUTTON INFORMATION

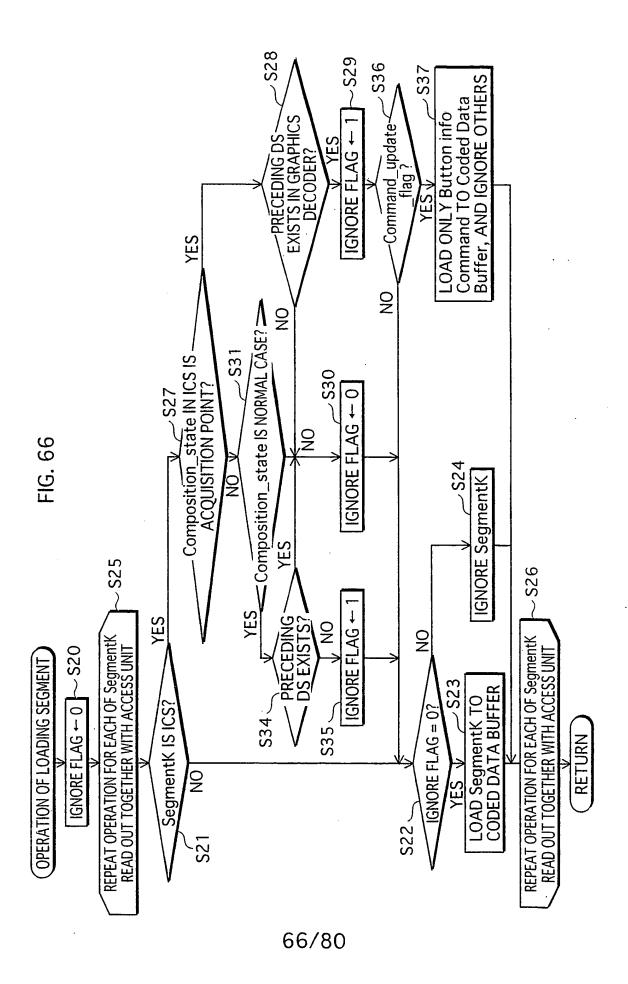


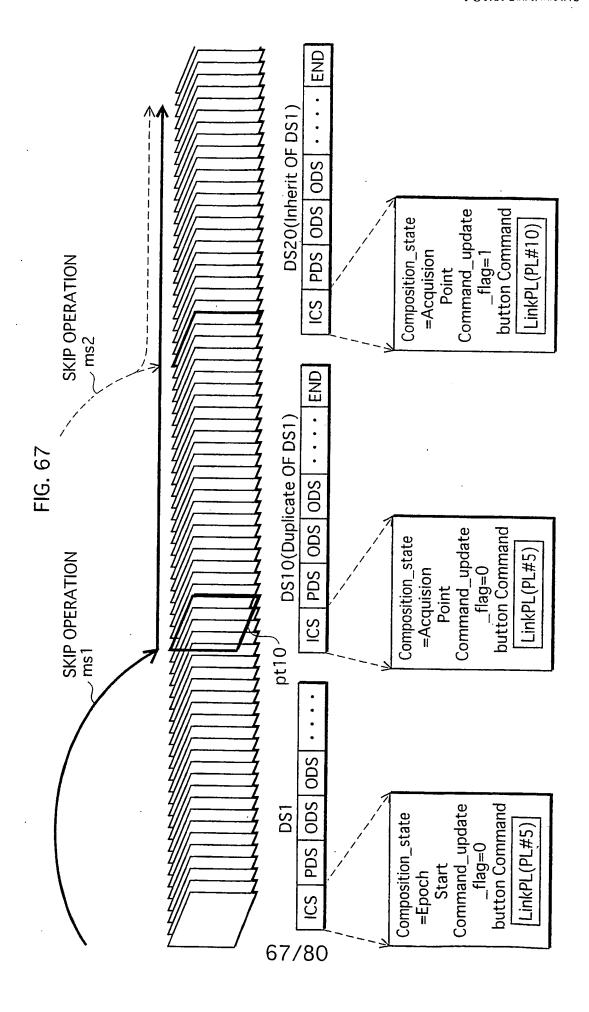


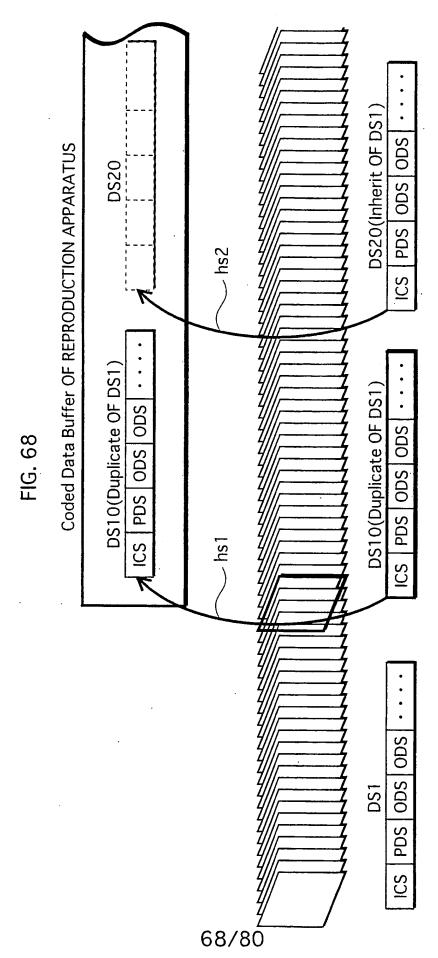


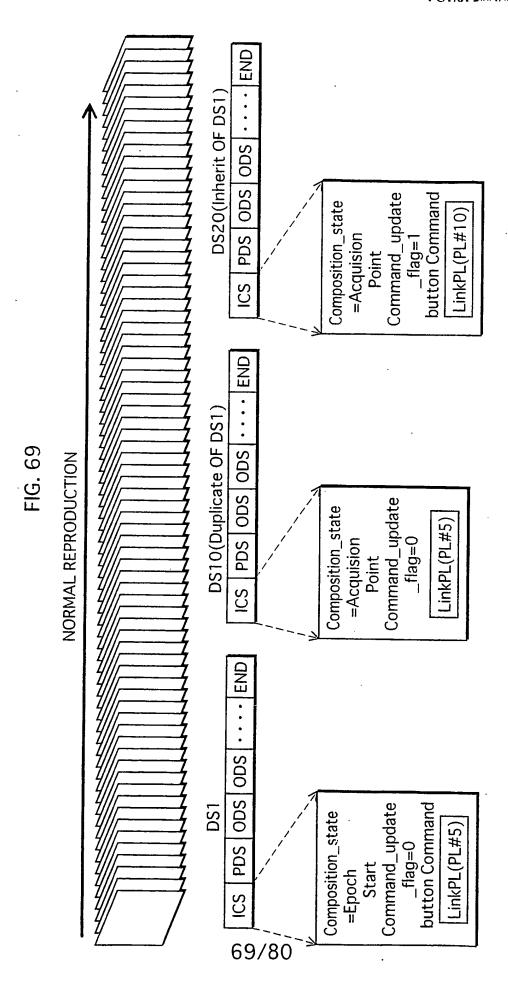
64/80











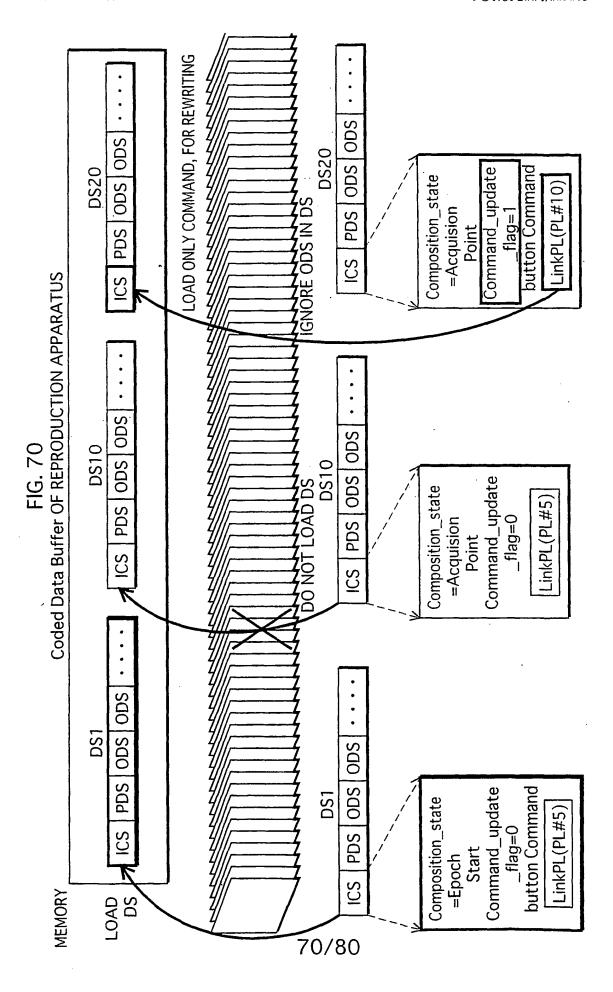
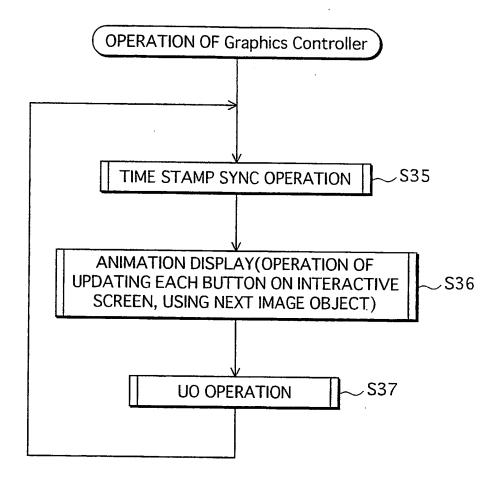
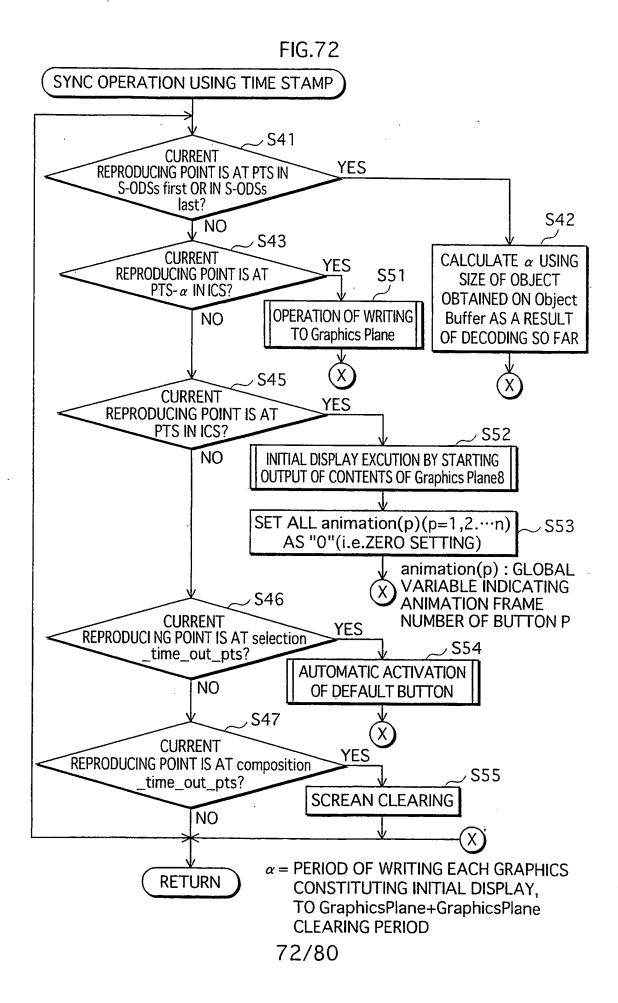


FIG.71



PCT/JP2004/009873



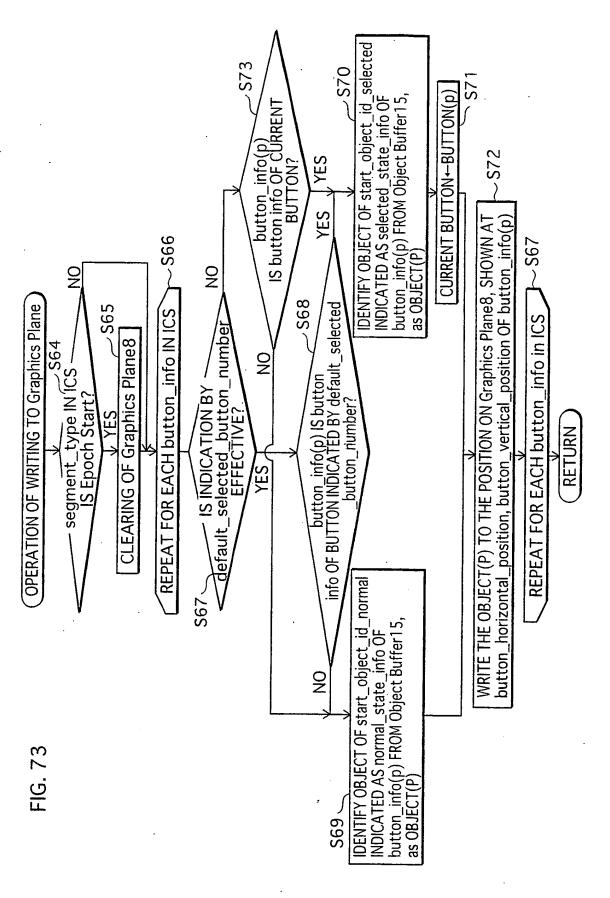
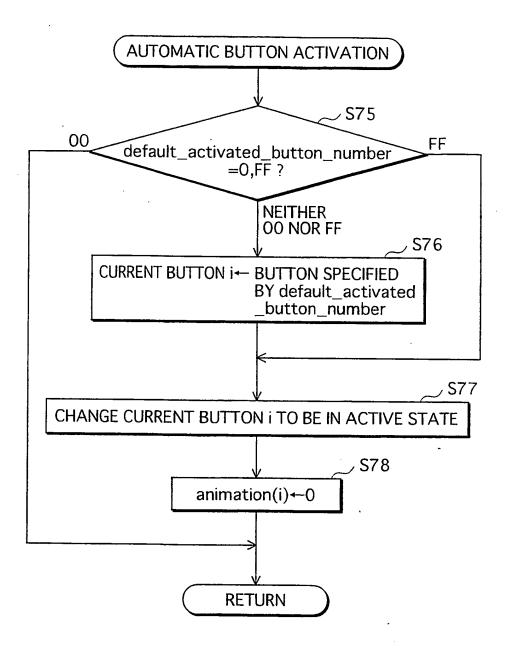
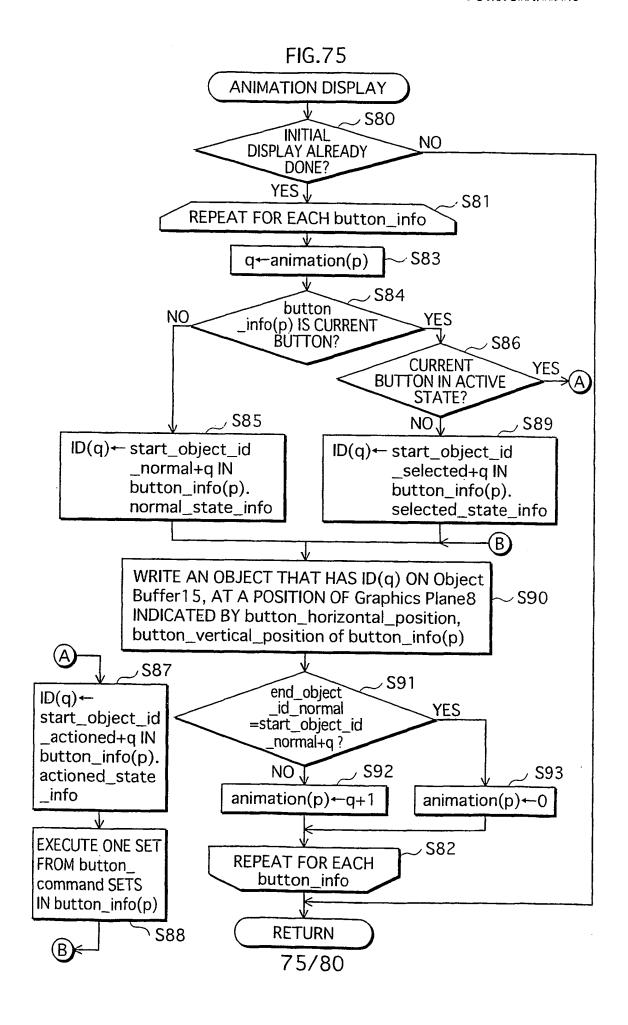


FIG.74





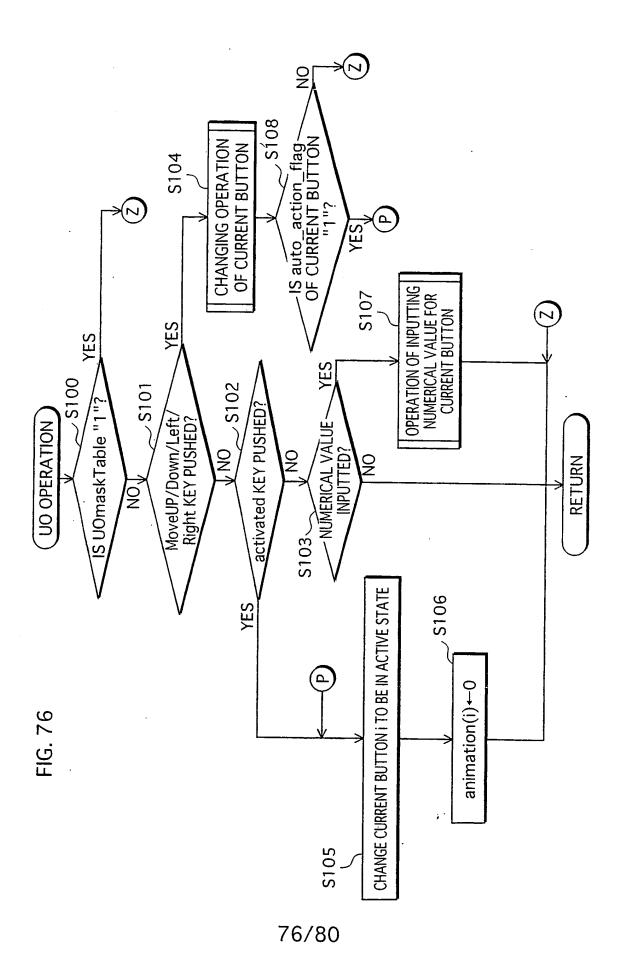


FIG.77

